

BAB II

TINJAUAN PUSTAKA

2.1 Profil dan Sejarah SD Khadijah 2 Surabaya

Secara singkat, sejarah berdirinya SD Khadijah 2 Surabaya dapat dijelaskan sebagai berikut. SD Khadijah 2 Surabaya merupakan sebagai salah satu sekolah dasar Islam yang diselenggarakan dan dibawah naungan YPIDP (Yayasan Pendidikan Islam Darmo Permai) sejak tahun 1985 dan di kelola oleh Kepala Sekolah Achmad Rijanto, ST., MT. Badan pendiri SD Khadijah 2 Surabaya yaitu KKUIDP (Kelompok Kerukunan Umat Islam Darmo Permai) dan KPWIDP (Kelompok Pengajian Wanita Islam Darmo Permai).

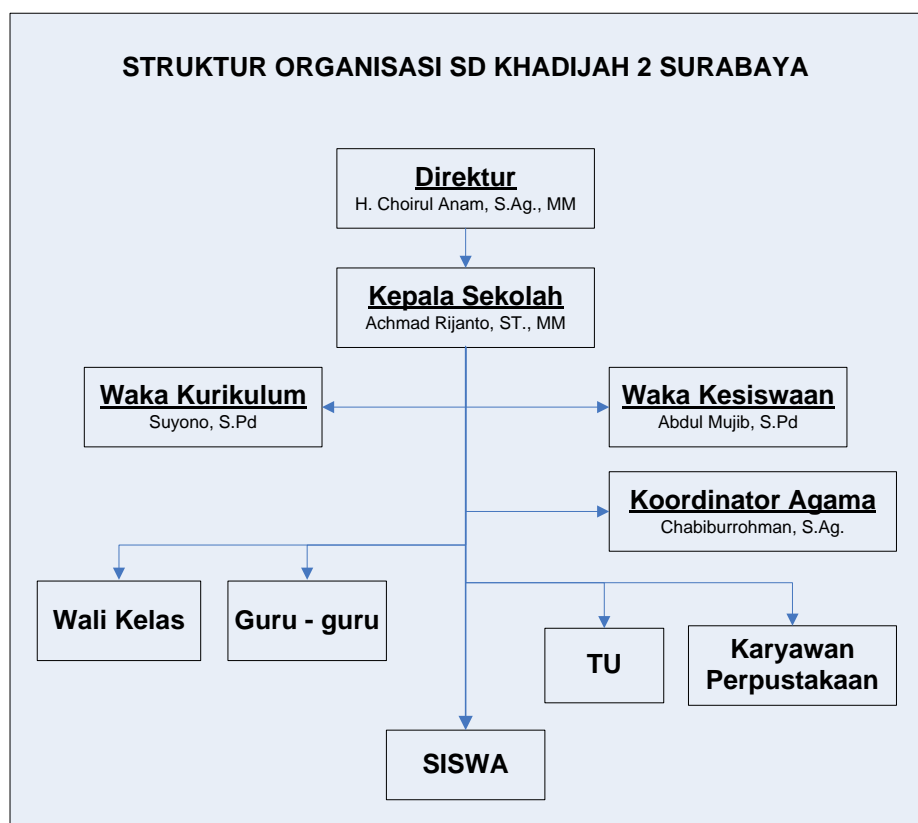
SD Khadijah 2 ini merupakan salah satu sekolah dasar Islam yang memiliki status SBI (Sekolah Berstandar Internasional). Sekolah Dasar Khadijah 2 ini memiliki banyak fasilitas untuk penunjang pendidikan diantaranya : ruang kelas yang berAC, memiliki perlengkapan peralatan laboratorium IPA, Bahasa dan computer, menerapkan strategi pembelajaran, aktif, kreatif, efektif, menyenangkan dan inovatif (PAKEMI), mempunyai peningkatan jumlah koleksi buku perpustakaan dalam kondisi baik, menerapkan budaya hari bahasa Internasional (Inggris dan Arab), mempunyai kurikulum yang mengacu pada kurikulum 2004 atau kurikulum 2006 dan masih banyak lagi.

2.1.1. Lokasi SD Khadijah 2 Surabaya

SD Khadijah 2 Surabaya ini berdiri di atas lahan sekitar 5.000 m² terletak di kawasan perumahan Darmo Permai Selatan V/61-63 Surabaya. Letak tempat sekolah yang strategis di dekat perumahan warga ini, membuat Sekolah Dasar ini menjadi Sekolah Dasar Islam favorit yang banyak dicari para wali murid. Dan juga tidak jauh dari Masjid dan jalan raya.

2.1.2. Struktur Organisasi

Gambar 2.1 merupakan dasar struktur organisasi yang ada di SD Khadijah 2 Surabaya.



Gambar 2.1. Struktur Organisasi SD Khadijah 2 Surabaya

2.1.3. Prosedur Pendaftaran Siswa Baru

Setiap tahun, SD Khadijah 2 Surabaya selalu membuka pendaftaran dan penerimaan siswa baru bagi calon siswa yang sudah lulus Taman Kanak-kanak / PlayGroup. Dalam hal ini, calon siswa baru harus membeli formulir pendaftaran yang tersedia di SD Khadijah 2 Surabaya dengan biaya formulir sebesar Rp. 200.000,- per formulir. Serta diharuskan untuk mengisi dan melengkapi persyaratan pendaftaran di antaranya :

- a. Membeli formulir pendaftaran sebesar Rp. 200.000,- per formulir.
- b. Mengisi formulir pendaftaran dengan benar.
- c. Calon siswa baru harus mengikuti Tes kepribadian / wawancara di SD Khadijah 2 Surabaya
- d. Usia calon siswa minimal 5-10 tahun.
- e. Menyerahkan foto copy Surat Keterangan Tamat Belajar (SKTB) bila ada / Buku Laporan Pribadi semester I (BLP).
- f. Menyerahkan foto 3 x 4 (2 lembar).

Setelah persyaratan pendaftaran telah dipenuhi, maka prosedur selanjutnya calon siswa baru diharuskan melakukan pembayaran Uang Bhakti / Infaq diantaranya sebagai berikut :

- a. Untuk calon siswa gelombang I biaya infaq sebesar Rp. 4.000.000,- sedangkan untuk calon siswa gelombang II biaya infaq sebesar Rp. 4.500.000,-
- b. Membayar uang Syahriyah (SPP) perdana sebesar Rp.300.000,-.

2.1.4. Prosedur Pembayaran Syahriyah (SPP)

Setiap bulan, siswa yang telah terdaftar menjadi siswa SD Khadijah 2 Surabaya, wajib membayar biaya Syahriyah (SPP) sebesar Rp. 300.000,-. Berikut merupakan prosedur siswa yang akan melakukan pembayaran Syahriyah :

- a. Apabila siswa yang ingin membayar Syahriyah, siswa wajib mengisi form bank yang tersedia di bagian administrasi dan menyerahkan buku tabungan kepada pihak administrasi.
- b. Pihak administrasi. (juru bayar) menyerahkan form yang telah terisi kepada pihak Bank BTN.
- c. Pihak Bank BTN memproses pendebitan per tanggal 20 dan menyerahkan kembali pada pihak administrasi.
- d. Bendahara yayasan akan membuat laporan pembayaran Syahriyah pada siswa yang telah membayar dan yang belum membayar.

2.2 Sistem Informasi

Sistem informasi adalah aplikasi computer untuk mendukung operasi dari suatu organisasi : operasi, instalasi, dan perawatan computer, perangkat lunak, dan data. Definisi Lain dari Sistem Informasi antara lain :

- a. Sistem informasi adalah sekumpulan hardware, software, brainware, prosedur atau aturan yang diorganisasikan secara integral untuk mengolah data menjadi informasi yang bermanfaat guna memecahkan masalah dan pengambilan keputusan.

- b. Sistem informasi adalah satu kesatuan data olahan yang terintegrasi dan saling melengkapi yang menghasilkan output baik dalam bentuk gambar, suara maupun tulisan.
- c. Sistem informasi adalah proses yang menjalankan fungsi mengumpulkan, memproses, menyimpan, menganalisis dan menyebarkan informasi untuk kepentingan tertentu.
- d. Sistem informasi adalah kumpulan antara sub-sub system yang saling berhubungan yang membentuk suatu komponen yang didalamnya mencakup input-proses-output yang berhubungan dengan pengolahan informasi (data yang telah diolah sehingga lebih berguna bagi user).
- e. Sistem informasi adalah sistem yang saling berhubungan dan terintegrasi satu dengan yang lain dan bekerja sesuai dengan fungsinya untuk mengatur masalah yang ada.[1]

2.3 Pengertian Website

Website atau *world wide web*, sering disingkat sebagai *www* atau "*web*" saja, yakni sebuah system dimana informasi dalam bentuk teks, gambar, suara, dan lain-lain dipresentasikan dalam bentuk *hypertext* dan dapat diakses oleh perangkat lunak yang disebut *browser*. Informasi di *web* pada umumnya ditulis dalam format HTML.

Keistimewaan inilah yang telah menjadikan *web* sebagai *service* yang paling cepat pertumbuhannya. *Web* mengizinkan pemberian *highlight* (penyorotan) pada kata-kata atau gambar dalam sebuah dokumen untuk menghubungkan ke dokumen lain.

Dengan sebuah *browser* yang memiliki *Graphical User Interface (GUI)*, *link-link* dapat dihubungkan ke tujuannya dengan menunjuk *link* tersebut dengan *mouse* dan menekannya.

2.3.1. Web Statis

Web statis adalah *web* yang menampilkan informasi-informasi yang sifatnya statis (tetap). Disebut statis karena pengguna tidak dapat berinteraksi dengan *web* tersebut. Singkatnya, untuk mengetahui suatu *web* bersifat statis atau dinamis dapat dilihat dari tampilannya. Jika suatu *web* hanya berhubungan dengan halaman *web* lain dan berisi suatu informasi yang tetap maka *web* tersebut statis.

Pada *web* statis, pengguna hanya dapat melihat isi dokumen pada halaman *web* dan apabila diklik akan berpindah ke halaman *web* yang lain. Interaksi pengguna hanya sebatas dapat melihat informasi yang ditampilkan. *Web* statis biasanya merupakan HTML yang ditulis pada editor teks dan disimpan dalam bentuk html atau htm.

2.3.2. Web Dinamis

Web dinamis adalah *web* yang menampilkan informasi serta berinteraksi dengan pengguna. *Web* yang dinamis memungkinkan pengguna untuk berinteraksi menggunakan form sehingga dapat mengolah informasi yang ditampilkan. *Web* dinamis bersifat interaktif, tidak kaku, dan terlihat lebih indah.[1]

2.4. Unified Modelling Process (UML)

UML adalah sebuah bahasa yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan

sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan class dan operation dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek.

Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan syntax/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML syntax mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML diturunkan dari 3 notasi yang telah ada sebelumnya : Grady Booch - Object-Oriented Design (OOD), Jim Rumbaugh - Object Modeling Technique (OMT), dan Ivar Jacobson - Object-Oriented Software Engineering (OOSE). Didalam UML terdapat Use Case Diagram, Class Diagram, Sequence Diagram, Collaboration Diagram, dan Deployment Diagram.

2.5. HTML (*Hypertext Markup Language*)

Hypertext Markup Language (HTML) merupakan sebuah bahasa *scripting* yang berguna untuk menuliskan halaman *web*. Pada halaman *web*, HTML dijadikan sebagai bahasa *script* dasar yang berjalan bersama berbagai bahasa *scripting* pemrograman lainnya.

Tidak diperlukan suatu program editor khusus untuk menggunakan perintah-perintah HTML, misalnya dapat menggunakan Notepad, Edit Plus ataupun editor lain yang berbasis GUI (*Graphical User Interface*) seperti Microsoft Frontpage, dan yang akan digunakan adalah Macromedia Dreamweaver 8. dengan program ini, kode HTML tidak perlu diketik, karena semua perintahnya diwujudkan secara icon base. Tetapi bagi seorang pengembang aplikasi *web*, kemampuan penguasaan terhadap kode-kode HTML sangat diperlukan, sehingga sangatlah disarankan untuk menguasai kode perintah HTML, maka pergunakanlah editor teks (misalnya Notepad).

Adapun secara umum tugas-tugas yang dapat dilakukan oleh HTML adalah :

- a. Mengontrol tampilan dari *web page* dan isinya.
- b. Mempublikasikan *web* secara *online*.
- c. Membuat *online form* yang bisa digunakan untuk menangani pendaftaran buku tamu, dan transaksi *online* lainnya.
- d. Menambahkan objek-objek seperti *image*, *audio*, *video*, dan juga *applet* dalam dokumen HTML.

Browser merupakan *software* yang diinstal di mesin *client* yang berfungsi untuk menterjemahkan tag-tag HTML menjadi halaman *web*. *Browser* yang biasanya digunakan adalah internet explorer, nescape navigator dan banyak *browser browser* yang lain.[4]

2.6. Definisi PHP

PHP singkatan dari *PHP Hypertext Preprocessor* yang digunakan sebagai bahasa *script serve-side* dalam pengembangan *web* yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan *web* dapat dibuat dinamis sehingga *maintenance* situs *web* tersebut lebih mudah dan efisien. PHP merupakan *software Open-Source* yang disebar dan dilisensikan secara gratis serta dapat di *download* secara bebas dari situs resminya yaitu : <http://www.php.net>.

PHP yang ditulis dengan menggunakan bahasa C dapat dikembangkan sendiri dengan menambah fungsi-fungsi baru. PHP adalah bahasa yang sangat populer. Sebagian besar sintaks PHP mirip dengan sintaks dalam bahasa C dan Perl.

2.6.1. Kelebihan PHP

PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa *script* sejenis. PHP difokuskan pada pembuatan *script server-side*, yang bisa melakukan apa saja yang dapat dilakukan oleh CGI, seperti mengumpulkan data dari form, menghasilkan isi halaman *web* dinamis, dan kemampuan mengirim serta menerima *cookies* bahkan lebih daripada kemampuan CGI.

PHP dapat digunakan pada semua system operasi, antara lain Linux, Unix, Microsoft Windows, Macintosh. PHP juga mendukung banyak *Web Server*, seperti Apache, *Microsoft Internet Information Server* (MIIS), *Personal Web Server* (PWS) dan masih banyak lagi lainnya, bahkan PHP dapat bekerja sebagai suatu *CGI processor*.

PHP tidak terbatas pada hasil keluaran HTML (*HyperText Markup Languages*). PHP juga memiliki kemampuan untuk mengolah keluaran gambar,

file PDF, dan *movies flash*. PHP juga dapat menghasilkan teks seperti XHTML dan file XML lainnya.

2.6.2. Sintaks PHP

Sintaks program atau script PHP ditulis dalam apitan tanda khusus PHP. Ada empat macam pasangan tag PHP yang dapat digunakan untuk menandai blok script PHP :

```
<?php ...?>
<script language = "PHP"> ... </script>
<? ... ?>
<% ... %>
```

Cara 1 dan 2 merupakan cara yang paling umum digunakan sekalipun cara 3 tampak lebih praktis karena cara 3 tidak selalu diaktifkan pada konfigurasi *file* php. Ini yang terdapat pada direktori C:\Program Files\Apache Software Foundation\Apache2.2\PHP.

Cara 4 juga dimungkinkan sebagai kemudahan bagi yang sudah terbiasa dengan ASP (*Active Server Pages*). Namun, bila itu tidak dikenal, maka harus dilakukan pengaktifan pada *file* konfigurasi php ini. Pengaktifan yang dilakukan pada *file* php ini terdapat pada baris berikut :

Language options

Allow ASP -style <% %>tags

Asp_tags = off,

Ubah *Off* menjadi *On*, kemudian simpan dan restart kembali *web server*.^[4]

2.7. Rational Rose

Rational Rose merupakan sebuah perangkat pemodelan secara visual yang memiliki banyak kemampuan (powerful) untuk pembentukan sistem berorientasi obyek yang menggunakan Unified Modeling Language (UML). UML merupakan bahasa pemodelan yang dapat digunakan secara luas dalam pemodelan bisnis, pemodelan perangkat lunak dari semua fase pembentukan dan semua tipe sistem, dan pemodelan secara umum dari berbagai pembentukan / konstruksi yang memiliki dua perilaku yaitu baik statis maupun dinamis. Rational Rose 2000, memiliki beberapa diagram yaitu :

- a. **Use case diagram** menjelaskan manfaat system jika dilihat menurut pandangan orang yang berada diluar system (actor). Use case diagram dapat digunakan selama proses analisis untuk menangkap requirements system. Dan selama tahap design, use case diagram menetapkan perilaku system saat di implementasikan.
- b. **Class diagram** membantu kita memvisualisasi struktur kelas-kelas dari suatu system. Class diagram memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas didalam model design (dalam logical view) dari suatu system.
- c. **Statechart diagram**, kita dapat menggunakan statechart diagram untuk memodelkan perilaku dinamis satu kelas atau objek. Statechart diagram memperlihatkan urutan keadaan sesaat (state) yang dilalui sebuah objek, kejadian yang menyebabkan sebuah transisi dari satu state atau aktifitas. Statechart diagram khususnya digunakan untuk memodelkan taraf-taraf diskrit dari sebuah siklus hidup objek.

- d. **Activity diagram** Memodelkan alur kerja (workflow) sebuah proses bisnis dan urutan aktivitas dalam suatu proses. Diagram ini sangat mirip dengan sebuah flowchart karena kita dapat memodelkan sebuah alur kerja dari satu aktivitas ke aktivitas lainnya atau dari satu aktivitas ke keadaan sesaat (state). Juga sangat berguna ketika ingin menggambarkan perilaku paralel atau menjelaskan bagaimana perilaku dalam berbagai use case berinteraksi.
- e. **Sequence diagram** Menjelaskan interaksi obyek yang disusun dalam suatu urutan tertentu. Sequence diagram memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu didalam use case.
- f. **Collaboration diagram** Melihat pada interaksi dan hubungan terstruktur antar obyek. Tipe diagram ini menekankan pada hubungan (relationship) antar obyek, sedangkan sequence diagram menekankan pada urutan kejadian. Collaboration diagram digunakan sebagai alat untuk menggambarkan interaksi yang mengungkapkan keputusan mengenai perilaku sistem.
- g. **Component diagram** Menggambarkan alokasi semua kelas dan obyek kedalam komponen-komponen dalam desain fisik sistem software. Diagram ini memperlihatkan pengaturan dan kebergantungan antara komponene-komponen software seperti source code, binary code dan komponen tereksekusi.
- h. **Deployment diagram** Diagram ini memperlihatkan pemetaan software kepada hardware. Diagram ini menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, dimana komponen akan terletak,

bagaimana kemampuan jaringan pada lokasi tersebut dan hal lain yang bersifat fisik (Sri Dharwiyanti dan Romi satria Wahono, 2003).

2.8. MySQL

MySQL adalah salah satu jenis *database server* yang sangat terkenal. Kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. Selain itu, ia bersifat *free* (tidak perlu membayar untuk menggunakannya) pada berbagai *platform* (kecuali pada Windows, yang bersifat *shareware* atau perlu membayar setelah melakukan evaluasi dan memutuskan untuk digunakan untuk keperluan produksi). Perangkat lunak MySQL sendiri bisa di *download* dari <http://www.mysql.org> atau <http://www.mysql.com>

MySQL termasuk jenis RDBMS (*Relation Database Management Sistem*). Pada MySQL, sebuah *database* mengandung satu atau sejumlah table. Table terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom. MySQL dapat didefinisikan sebagai :

- a. MySQL merupakan system manajemen *database*. *Database* merupakan struktur penyimpanan data. Untuk menambah, mengakses, dan memproses data yang disimpan dalam sebuah *database* computer, diperlukan system manajemen *database MySQL server*.
- b. MySQL merupakan system manajemen database atau basis data terhubung (*Relation Database Management System*). *Database* terhubung menyimpan data pada table-tabel terpisah. Kata SQL pada MySQL merupakan singkatan dari “*Structure Query Language*”. SQL

merupakan bahasa standar yang digunakan untuk mengakses *database* dan ditetapkan oleh ANSI/ISO SQL Standard.

- c. MySQL merupakan *Software open Source*. *Open Source* berarti semua orang diizinkan menggunakan dan memodifikasi *software*. Semua orang dapat mendownload *software* MySQL dari internet dan menggunakannya tanpa membayar. Serta dapat dengan mudah mempelajari *Source Code* dan akan menggunakannya sesuai kebutuhan.
- d. Server *database* MySQL mempunyai kecepatan akses tinggi, mudah digunakan dan handal. MySQL dikembangkan untuk menangani *database* yang besar secara cepat dan telah sukses digunakan selama bertahun-tahun. Konektifitas, kecepatan, dan keamanannya membuat *server* MySQL cocok untuk mengakses *database* di internet.
- e. MySQL *Server* bekerja di *client/server* atau *system embedded*. *Software database* MySQL merupakan *system client/server* yang terdiri atas *multithread SQL server* yang mendukung *software client* dan *library* yang berbeda, *tool administrative*, dan sejumlah *Application Programming Interfaces* (API).

Secara umum, SQL terdiri dari dua bahasa, yaitu *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML). Implementasi DDL dan DML berbeda untuk tiap sistem manajemen basis data (SMBD), namun secara umum implementasi tiap bahasa ini memiliki bentuk standar yang ditetapkan ANSI. Artikel ini akan menggunakan bentuk paling umum yang dapat digunakan pada kebanyakan SMBD.

DDL digunakan untuk mendefinisikan, mengubah, serta menghapus basis data dan objek yang diperlukan dalam basis data, misalnya tabel, view, user, dan sebagainya. Secara umum, DDL yang digunakan adalah CREATE untuk membuat objek yang baru, USE untuk menggunakan objek, ALTER untuk mengubah objek yang sudah ada, dan DROP untuk menghapus objek. DDL biasanya digunakan oleh administrator basis data dalam pembuatan sebuah aplikasi basis data.[3]

Untuk membuat tabel pada SQL dilakukan dengan memasukkan query sebagai berikut :

```
CREATE TABLE [nama_tabel]
(
  nama_field1 tipe_data [,
  nama_field2 tipe_data,
  ...]
  [CONSTRAINT nama_field constraints]
)
```

2.9. Apache

Apache merupakan *software* yang dikeluarkan oleh Group *Aphace*. Group *Apache* terlibat dalam suatu proyek yang disebut Proyek *Apache* untuk mengembangkan suatu *software* implementasi di *server* HTTP (*web*) yang handal, standar komersial, dan *Source Code*-nya didistribusikan secara gratis. Adapun alamat *website Apache* adalah : <http://www.Apache.org/>.

2.10. JavaScript

JavaScript mulanya bernama *LiveScript*, dikembangkan pertama kali tahun 1995 di *Netscape Communication*. Pada akhir tahun 1995 *Netscape Communication* dan *Sun Microsystem* berkolaborasi dan mengganti nama *LiveScript* menjadi *JavaScript*.

JavaScript adalah bahasa skrip yang ditempatkan pada kode HTML dan diproses pada sisi klien. Dengan adanya bahasa ini maka kemampuan dokumen HTML menjadi lebih luas. Sebagai contoh, digunakan untuk validasi masukan pada formulir sebelum diproses ke tahap selanjutnya. Bisa untuk membuat permainan interaktif dan juga bisa untuk menambah desain *web*.

2.11. Ajax

Ajax adalah singkatan dari Asynchronous JavaScript and XML. Pada dasarnya ajax menggunakan XMLHttpRequest object Javascript untuk membuat request ke server secara asynchronous atau tanpa melakukan refresh halaman website. Yang dibutuhkan agar ajax dapat berjalan adalah javascript harus di enable pada browser yang digunakan. Walaupun javascript merupakan dasar dari Ajax, dimana javascript sangat susah pada implementasi dan maintenance, tetapi Ajax memiliki struktur pemrograman yang lebih mudah untuk dipahami. Membuat object XMLHttpRequest dan memastikan object tersebut terbentuk dengan benar. Kemudian menentukan kemana hasilnya akan ditampilkan atau dikirim.

Ajax dapat digunakan untuk melakukan banyak hal, seperti loading halaman HTML tanpa refresh halaman web, validasi form dan banyak lagi yang bisa dilakukan dengan ajax. Ajax bertugas melakukan request ke web server dan PHP yang berada di server akan melakukan apa yang diminta oleh Ajax, mengirim hasilnya ke web browser dan Ajax menampilkannya kepada user.

2.12. Power Designer

Satu paket permodelan dan perancangan untuk perusahaan yang memerlukan pembangunan atau perancangan ulang aplikasi bisnis dengan cepat, hemat dan konsisten. Power Designer terintegrasi untuk semua RDBMS modern dan pengembangan untuk Java™, .NET, PowerBuilder® dan Web Services. Seperangkat alat permodelan yang unik dengan menggabungkan beberapa teknik permodelan standar, yaitu: permodelan aplikasi melalui UML, Permodelan Proses Bisnis dan permodelan data yang paling banyak diminati pasar, semuanya didukung oleh solusi manajemen meta-data yang tangguh dan protektifitas yang sangat sempurna di bidangnya.

a. ERD

Entity Relationship Diagram adalah pemodelan data utama dan akan membantu mengorganisasikan data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas.

b. Entitas (Entity)

Entitas adalah suatu yang nyata atau abstrak dimana kita akan menyimpan data. Contoh : entitas pegawai, entitas kampus, entitas buku dsb.

c. Relasi (Relationship)

Relasi adalah hubungan alamiah yang terjadi antara satu atau lebih entitas, misal proses pembayaran pegawai. Kardinalitas menentukan kejadian suatu entitas untuk satu kejadian pada entitas yang berhubungan. Misal, mahasiswa bisa mengambil banyak mata kuliah.

d. Atribut (Attribute)

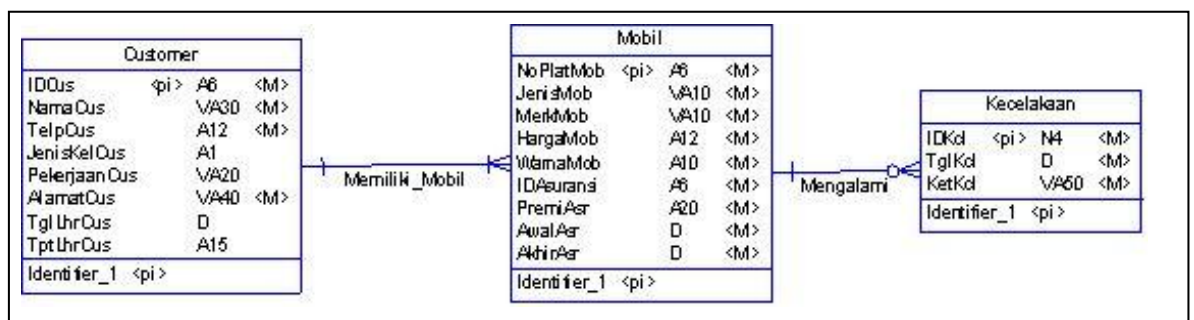
Atribut adalah ciri umum semua atau sebagian besar instansi pada entitas tertentu. Sebutan lain atribut adalah properti, elemen data dan field. Misal, nama, alamat, nomor pegawai, gaji dsb.

e. Metodologi ERD

- a. Menentukan entitas Menentukan peran, kejadian, lokasi, hal nyata dan konsep dimana penggunaan untuk menyimpan data.
- b. Menentukan relasi Menentukan hubungan antar pas angan entitas menggunakan matriks relasi.
- c. Gambar ERD sementara Entitas digambarkan dengan kotak, dan relasi digambarkan dengan garis.
- d. Isi kardinalitas Menentukan jumlah kejadian satu entitas untuk sebuah kejadian pada entitas yang berhubungan.
- e. Tentukan kunci utama Menentukan atribut yang mengidentifikasi satu dan hanya satu kejadian masing-masing entitas.
- f. Gambar ERD berdasarkan kunci Menghilangkan relasi many to many dan memasukkan primary dan kunci tamu pada masing masing entitas.
- g. Menentukan atribut Menentukan field -field yang diperlukan system.
- h. Pemetaan atribut Memasangkan atribut dengan entitas yang sesuai.
- i. Gambar ERD dengan atribut Mengatur ERD dari langkah 6 dengan menambahkan entitas atau relasi yang ditemukan pada langkah 8.
- j. Periksa hasil Apakah ERD sudah menggambarkan system yang akan dibangun.

f. CDM

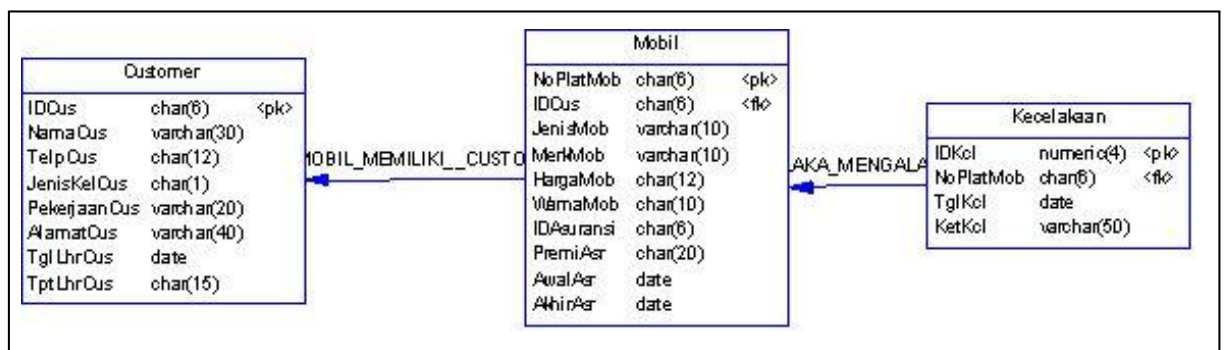
Conceptual Data Model (CDM) merupakan model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas (*entity*) serta hubungan (*relationship*) antara entitas-entitas itu.



Gambar 2.2. Contoh Conceptual Data Modeling

g. PDM

Physical Data Model (PDM) merupakan model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap tabel mempunyai sejumlah kolom di mana setiap kolom memiliki nama yang unik.



Gambar 2.3. Contoh Physical Data Modeling

2.13. Data Manipulation Language

DML digunakan untuk memanipulasi data yang ada dalam suatu tabel.

Perintah yang umum dilakukan adalah:

- a. SELECT untuk menampilkan data
- b. INSERT untuk menambahkan data baru
- c. UPDATE untuk mengubah data yang sudah ada
- d. DELETE untuk menghapus data

SELECT adalah perintah yang paling sering digunakan pada SQL, sehingga terkadang istilah *query* dirujuk pada perintah SELECT. SELECT digunakan untuk menampilkan data dari satu atau lebih tabel, biasanya dalam sebuah basis data yang sama. Secara umum, perintah SELECT memiliki bentuk lengkap:

```
SELECT [nama_tabel|alias.]nama_field1 [AS alias1] [, nama_field2,
...]
FROM nama_tabel1 [AS alias1] [INNER|LEFT|RIGHT JOIN tabel2 ON
kondisi_penghubung]
[, nama_tabel3 [AS alias3], ...]
[WHERE kondisi]
[ORDER BY nama_field1 [ASC|DESC][, nama_field2 [ASC|DESC], ...]]
[GROUP BY nama_field1[, nama_field2, ...]]
[HAVING kondisi_aggregat]
```

dengan:

- a. kondisi adalah syarat yang harus dipenuhi suatu data agar ditampilkan.
- b. kondisi_aggregat adalah syarat khusus untuk fungsi agregat.

Kondisi dapat dihubungkan dengan operator logika, misalnya AND, OR, dan sebagainya.

Contoh 1: Tampilkan seluruh data.

```
SELECT *  
FROM user
```

Contoh 2: Tampilkan pengguna yang tidak pernah bertransaksi.

```
SELECT *  
FROM user  
WHERE total_transaksi = 0
```

Contoh 3: Tampilkan username pengguna yang bertransaksi kurang dari 10 dan nilainya lebih dari 1.000.

```
SELECT username  
FROM user  
WHERE jml_transaksi < 10 AND total_transaksi > 1000
```

Contoh 4: Tampilkan total nominal transaksi yang sudah terjadi.

```
SELECT SUM(total_transaksi) AS total_nominal_transaksi  
FROM user
```

Contoh 5: Tampilkan seluruh data diurutkan berdasarkan jumlah transaksi terbesar ke terkecil.

```
SELECT *  
FROM user  
ORDER BY jml_transaksi DESC
```

2.13.1. Fungsi Agregat

Beberapa SDBD memiliki fungsi agregat, yaitu fungsi-fungsi khusus yang melibatkan sekelompok data (agregat). Secara umum fungsi agregat adalah:

- a. SUM untuk menghitung total nominal data
- b. COUNT untuk menghitung jumlah kemunculan data
- c. AVG untuk menghitung rata-rata sekelompok data
- d. MAX dan MIN untuk mendapatkan nilai maksimum/minimum dari sekelompok data.

Fungsi agregat digunakan pada bagian SELECT. Syarat untuk fungsi agregat diletakkan pada bagian HAVING, bukan WHERE.[5]

2.13.2. Subquery

Ada kalanya *query* dapat menjadi kompleks, terutama jika melibatkan lebih dari satu tabel dan/atau fungsi agregat. Beberapa SDBD mengizinkan penggunaan *subquery*. Contoh :

Tampilkan username pengguna yang memiliki jumlah transaksi terbesar.

```
SELECT username
FROM user
WHERE jml_transaksi =
(
SELECT MAX(jml_transaksi)
FROM user
)
```

2.13.3. Insert

Untuk menyimpan data dalam tabel dipergunakan sintaks:

```
INSERT INTO [NAMA_TABLE] ([DAFTAR_FIELD]) VALUES ([DAFTAR_NILAI])
```

Contoh:

```
INSERT INTO TEST (NAMA, ALAMAT, PASSWORD) VALUES ('test',  
'alamat', 'pass');
```

2.13.4. Update

Untuk mengubah data menggunakan syntax:

```
UPDATE [NAMA_TABLE] SET [NAMA_KOLOM]=[NILAI] WHERE [KONDISI]
```

Contoh:

```
UPDATE Msuser set password="123456" where username="abc"
```

2.13.5. Delete

Untuk menghapus data dipergunakan sintaks:

```
DELETE FROM [NAMA_TABLE] [KONDISI]
```

Contoh:

```
DELETE FROM TEST WHERE NAMA='test';
```

BAB III

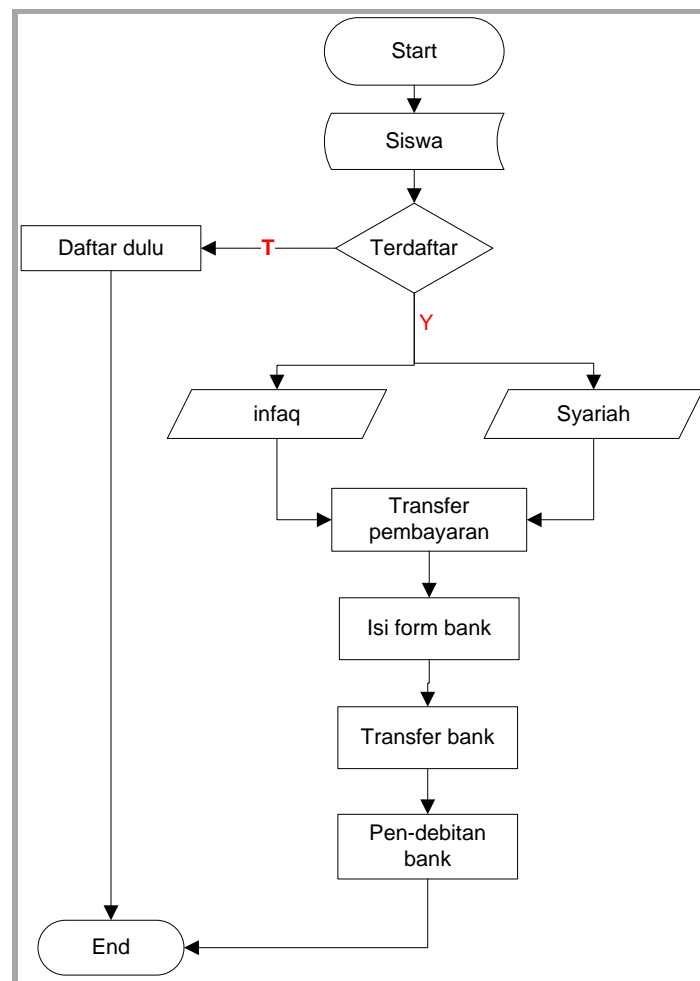
ANALISIS DAN PERANCANGAN PROGRAM

3.1 Analisis dan Konsep Perancangan Program

Analisis adalah penguraian dari suatu masalah atau objek yang akhirnya menghasilkan suatu kesimpulan, hal ini dimaksudkan untuk mengidentifikasi dan mengevaluasi masalah-masalah atau objek.

Analisis terhadap pembuatan program *website* ini dimaksudkan untuk mengetahui proses pembuatan program aplikasi *website* sehingga siap dipublikasikan ke internet. Maka dari diberikan teknik dasar dalam perancangan program aplikasi *website* secara umum, yaitu :

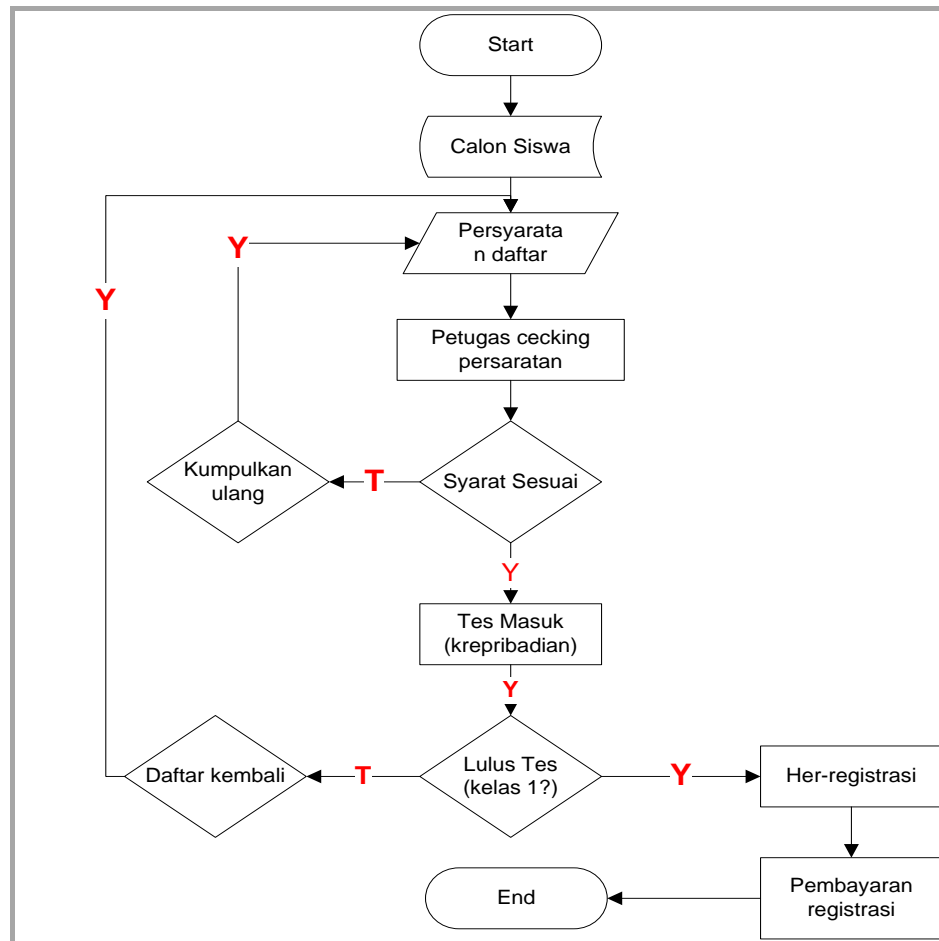
- a. Mengetahui *port* yang digunakan untuk melakukan koneksi ke *server* dalam aplikasi *website*.
- b. Menentukan bahasa pemrograman *web* yang akan digunakan.
- c. Menentukan fasilitas tambahan yang akan diberikan pada aplikasi *website*.
- d. Jika fasilitas tambahan membutuhkan database, maka tentukan *database* yang akan digunakan, sesuaikan dengan pemrograman *web* yang akan digunakan.
- e. Karena aplikasi *website* dapat dilihat oleh seluruh masyarakat internet maka gunakan desain aplikasi yang menarik dan mudah digunakan.
- f. Karena *website* dapat diakses secara umum, maka perlu menentukan hak akses pada tiap admin.



Gambar 3.1. Flowchart Proses Pembayaran SPP

Dari gambar yang ada diatas maka dapat dideskripsikan proses yang menjadi alur dimana saat siswa melakukan pembayaran, sebelum melakukan pembayaran maka dilakukan pengecekan apakah siswa tersebut telah terdaftar atau belum jika belum maka siswa tidak dapat melakukan proses ini, Pembayaran dilakukan langsung kepada kasir sekolah, yaitu melalui proses pengecekan, pemberian pembayaran, sampai munculnya Laporan bendahara. Terdapat dua pembayaran disini yaitu *Pertama* Pembayaran infaq, pembayaran siswa pada saat setelah calon siswa diterima menjadi siswa. *Kedua* Pembayaran

syariah, adalah iuran wajib bagi siswa yang dibayarkan per-bulan sejumlah yang ditentukan pihak sekolah.



Gambar 3.2. *Flowchart* Proses Pendaftaran

Flowchart tersebut menggambarkan tentang proses pendaftaran yang dilakukan oleh calon siswa dari pengumpulan data, tes kepribadian hingga pada registrasi setelah calon siswa tersebut lulus tes kepribadian. Untuk lebih spesifiknya dapat dijabarkan sebagai berikut :

- Pengumpulan persyaratan pendaftaran.
- Cek kelengkapan persyaratan dengan ketentuan yang ada, jika telah sesuai maka calon siswa dapat melakukan proses selanjutnya jika tidak maka

calon siswa tersebut diberi hak untuk dapat mengumpulkan kembali persyaratan tersebut.

- c) Setelah sesuai maka calon siswa dapat melakukan tes kepribadian, jika benar maka siswa dapat melakukan proses registrasi dengan mengisi form dan melakukan pembayaran.
- d) Proses registrasi, untuk mematenkan bahwa calon siswa tersebut telah menjadi siswa yang telah terdaftar.

3.2 Analisis Kebutuhan

Mencakup hal-hal yang dibutuhkan yang kesemuanya meliputi kebutuhan-kebutuhan pendukung pada pembuatan sistem hingga selesai yaitu berfokus pada alat yang dibutuhkan baik perangkat keras maupun lunak dan kebutuhan data yang meliputi data-data yang dibutuhkan yang berhubungan dengan pengguna sistem.

3.2.1 Identifikasi Kebutuhan Pendukung

Kebutuhan disini meliputi alat-alat yang dapat mendukung dalam merancang dan pembuatan sistem informasi administrasi tersebut.

Software dan Hardware yang dibutuhkan antara lain :

- a. *Power Designer* 6 (untuk merancang CFD & DFD) & *Power Designer* 11 (untuk merancang CDM & PDM)
- b. *PHP MyAdmin* (Apache Web Server, Database MySQL, PHP Script)
- c. *Macromedia Dreamweaver* 8
- d. *Photoshop CS3* (Edit Gambar)
- e. Processor minimum 200 MHz (sebaiknya lebih)

- f. RAM minimum 256 MB atau lebih
- g. VGA Card minimal 4MB.
- h. Cache memory 512 KB atau lebih
- i. Harddisk 40 GB.
- j. Monitor SVGA resolusi 800 x 600, 256 warna (minimal).
- k. Keyboard dan mouse

3.2.2 Kebutuhan Data

Dari hasil surfey yang dilakukan penulis maka dapat didefinisikan beberapa kebutuhan data atau requirement yang diperlukan dalam pembuatan aplikasi ini dan itu meliputi :

Tabel 3.1. *Data Requirement*

DATA		KAMUS DATA
Admin	=	Orang yang mempunyai wewenang/ kewajiban untuk mengelolah web.
Data Login Administrator	=	Nama dan Password yang dimiliki
Input Registrasi	=	Data input siswa setelah melakukan pendaftaran siswa baru
Bukti Pembayaran	=	Data yang berhubungan dengan administrasi pembayaran yang dilakukan oleh yang bersangkutan.
Data Formulir Bank	=	Pengisian dilakukan pada saat Orang Tua/Wali/Siswa hendak melakukan pembayaran melalui bank
Data Formulir Pendaftaran Siswa Baru	=	Data-data yang digunakan untuk identifikasi siswa yang mendaftar yang berisi data-data siswa yang sangat diperlukan.
Data Keuangan	=	Data dari hasil perhitungan iuran-iuran yang dibebankan kepada siswa

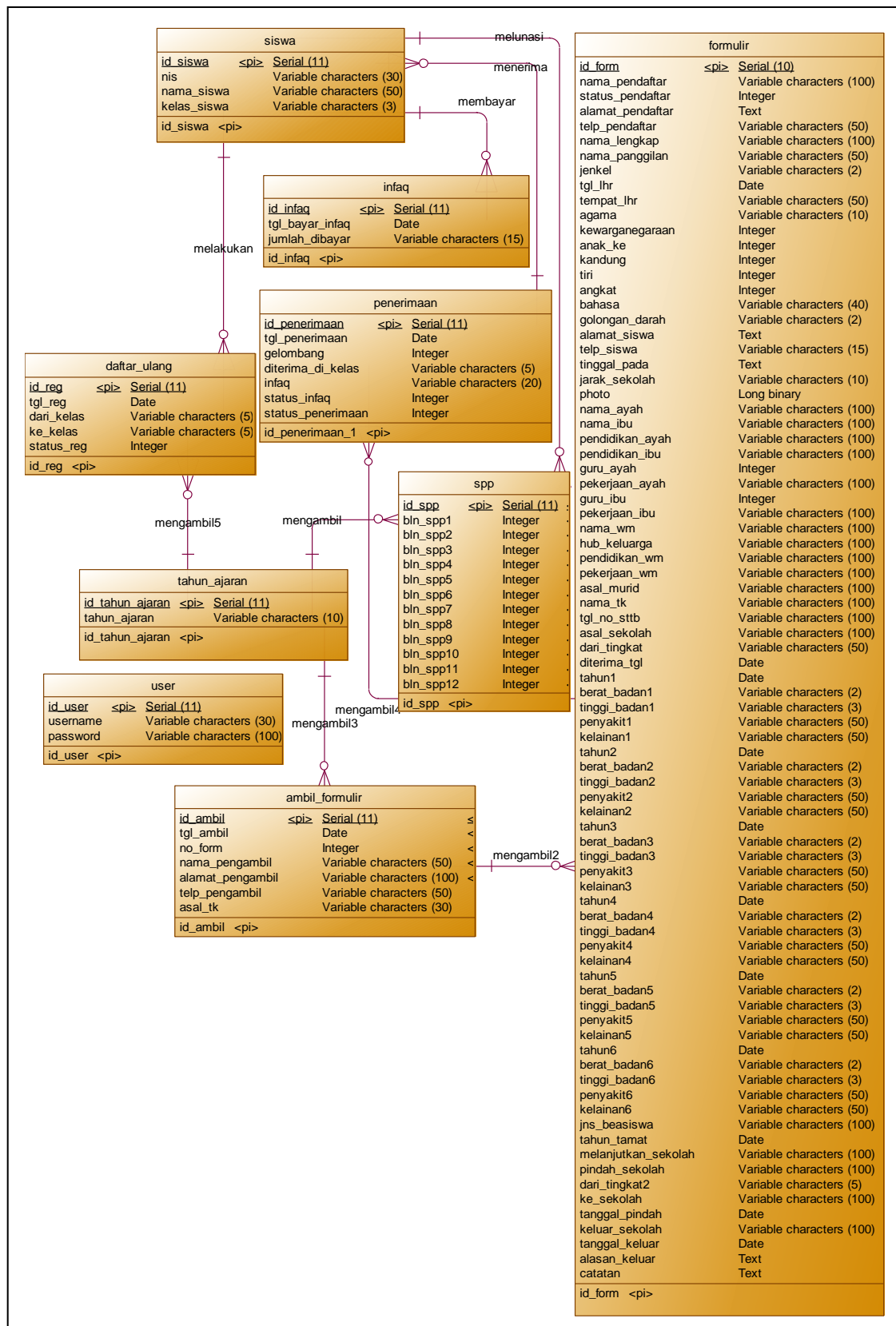
Kesemua data-data tersebut (lihat tabel data *requirement*) tersimpan dalam tabel-tabel yang terangkum dalam satu database yang telah dibuat, jadi

setiap data yang dimasukkan oleh user tersebut secara otomatis akan tersimpan pada database seperti yang terlihat pada CDM dan PDM .

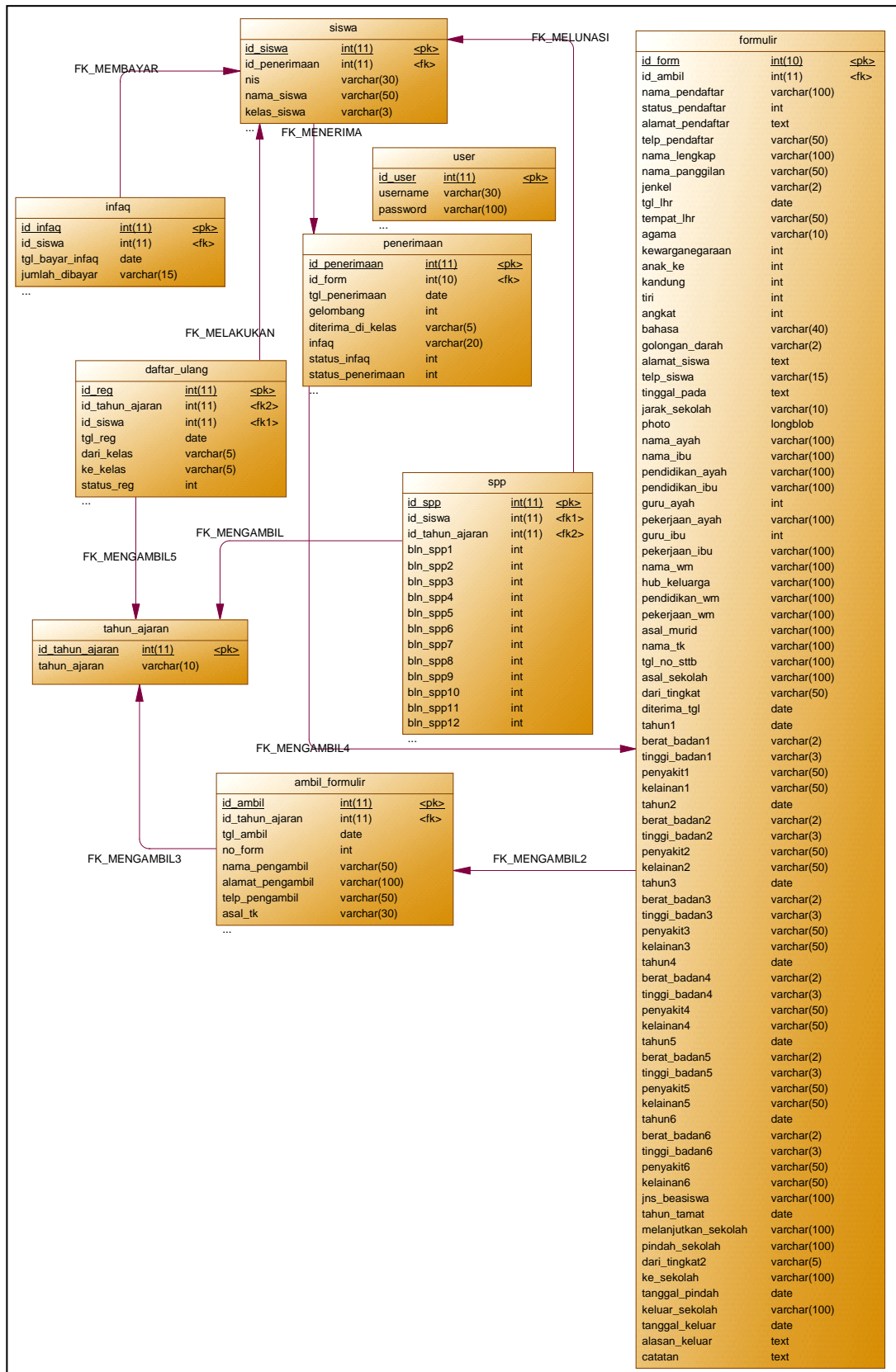
Conceptual Data Model (CDM) memodelkan struktur logis dari keseluruhan aplikasi data, tidak tergantung pada *software*. *Physical Data Model* (PDM) memodelkan struktur fisik dari basis data (*database*), dengan mempertimbangkan perangkat lunak (*software*).

Setelah CDM dibuat maka selanjutnya adalah proses pembuatan PDM (*Physical Data Model*), hal ini dibuat menurut hasil dari pembuatan CDM yang di-generate sehingga menjadi bentuk PDM yang selanjutnya berfungsi agar dapat di generate kedalam database SQL dan hasilnya digunakan untuk menciptakan susunan tabel yang ada dalam database.

Physical Data Model (PDM) memodelkan struktur fisik dari basis data (*database*), dengan mempertimbangkan perangkat lunak (*software*) DBMS serta model struktur yang akan digunakan. Untuk selanjutnya agar lebih menjelaskan dan menjabarkan tabel-tabel yang ada pada database, dapat dilihat pada tampilan Gambar 3.3 dan Gambar 3.4.



Gambar 3.3. CDM (Conceptual Data Model)



Gambar 3.4. PDM (Physical Data Model)

3.3 Perancangan Program Website

Struktur program aplikasi ini dibuat secara modular, yaitu program dipecah menjadi modul-modul kecil yang mudah dibuat, mudah dites, mudah melacak kesalahan program, dan mudah dimodifikasi. Rancangan sistem menu yang digunakan untuk membuat program aplikasi ini adalah sebagai berikut :

3.4 Perancangan Sistem Dalam UML

Perancangan sistem aplikasi *web* ini dibuat rancangan dengandengan memnggunakan *Unified Modeling Language* perancangan UML dibagi menjadi beberapa tahapan sebagai berikut :

a. Use Case Diagram

Berfungsi menggambarkan fungsionalitas yang diharapkan sebuah sistem. Ditunjukkan dengan interaksi antara aktor (entitas manusia/mesin) dan proses yang terjadi dalam system

b. Activity Diagram

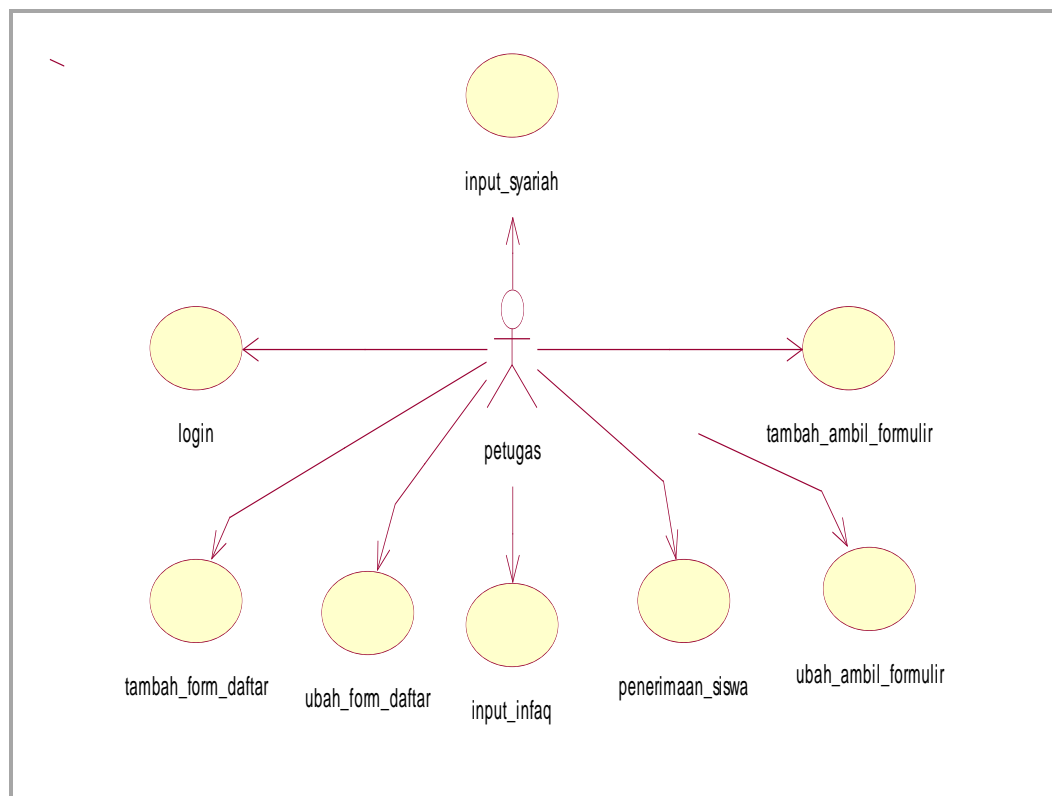
Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*).

c. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi *horizontal* (objek-objek yang terkait). *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara *internal* dan *output* apa yang dihasilkan.

3.4.1 Use Case Diagram

Pembuatan UML dalam perancangan pada sistem (*Unified Modeling Language*) dimulai dengan *Use Case Diagram* yang menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (*actor*). Use case diagram dapat digunakan selama proses analisis untuk menangkap requirements sistem. Dan selama tahap design, use case diagram menetapkan perilaku sistem saat di implementasikan. Penggambaran *Use Case Diagram* tersebut mempresentasikan perancangan sistem pada sekolah Khadijah secara umum yaitu sebuah interaksi antar aktor dan antar sistem pada instansi sekolah seperti pada Gambar 3.5.



Gambar 3.5. Pembuatan *Use Case Diagram*

Seorang aktor merupakan entitas manusia atau mesin yang berinteraksi dengan sistem sehingga dapat mencapai apa yang diharapkan oleh sistem. Aktor tersebut adalah petugas dalam hal ini bisa juga disebut admin yang bertugas layaknya administrator sistem *web* pada umumnya, beberapa proses sistem yang disebut disini adalah *input_syariah* (memasukkan data pembayaran syariah yang telah dilakukan oleh siswa/ortu), *login* (memasukkan *account* petugas untuk memasuki system), *tambah_ambil_formulir* (penambahan “**Pengambilan Formulir**”), *ubah_ambil_formulir* (pengubahan data yang telah ada pada “**Pengambilan Formulir**”), *input_infaq* (memasukkan data pembayaran infaq yang telah dilakukan oleh siswa/ortu) , *tambah_form_daftar* (penambahan data pada “**Formulir Pendaftaran**”), *ubah_form_daftar* (proses pengubahan data yang

telah ada pada “**Formulir Pendaftaran**”). Aktor dan sistem dihubungkan menurut kebutuhan masing-masing (lihat Gambar 3.5).

Terkait proses pembayaran dan pendaftaran pada keseluruhan *web* sistem yang dilakukan oleh seorang *siswa/ortu* pada sekolah melalui web sistem dan *petugas* akan dijelaskan secara komprehensif pada tahap UML selanjutnya dalam *activity diagram*.

3.4.2 Activity Diagram

Tatanan alur beberapa aktivitas dari setiap proses yang ada pada sistem sehingga hubungan pemakaian antara user dan sistem terjalin dengan baik (pemrosesan dalam sistem seperti yang diinginkan), adalah *activity diagram* ini yang dapat menjelaskan hubungan tersebut. Oleh karena user pemakai sistem hanya satu user, yaitu hanyalah petugas saja maka *activity diagram* yang ada adalah sebagai berikut :

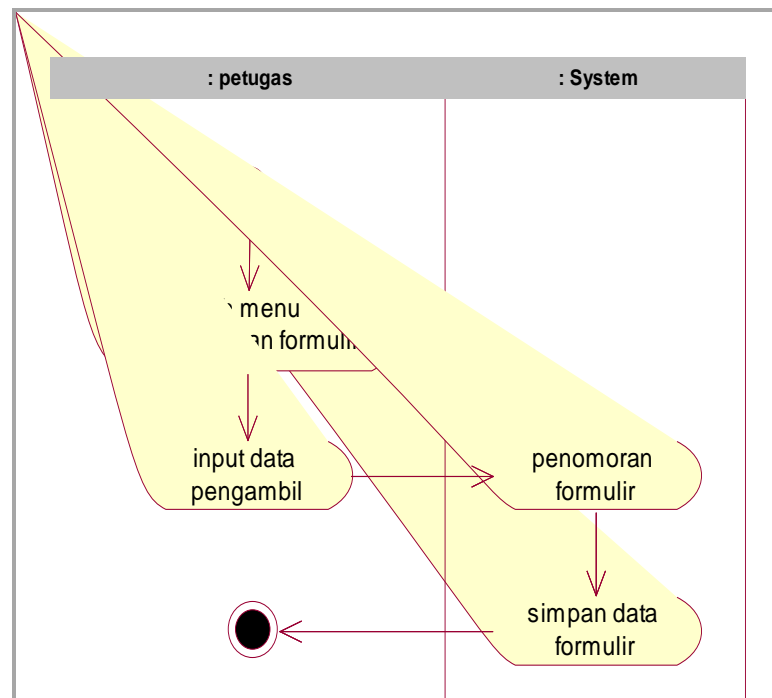
1) Pengambilan Formulir

Proses ini dilakukan setelah siswa menyerahkan formulir yang digunakan untuk melakukan tes masuk kepada petugas, formulir ini digunakan untuk pendataan calon siswa untuk melakukan tes masuk. Pada proses yang dilakukan terdapat dua aktivitas yang menyangkut segala kegiatan user (petugas) yang ada dan ada hubungannya dengan sistem, yaitu tambah formulir dan ubah data formulir yang semuanya disebutkan pada penjelasan dan gambar berikut ini :

a. Tambah Formulir

Adalah bagaimana seorang petugas untuk menambah calon siswa yang akan masuk mengikuti tes masuk (tes kepribadian). Proses yang

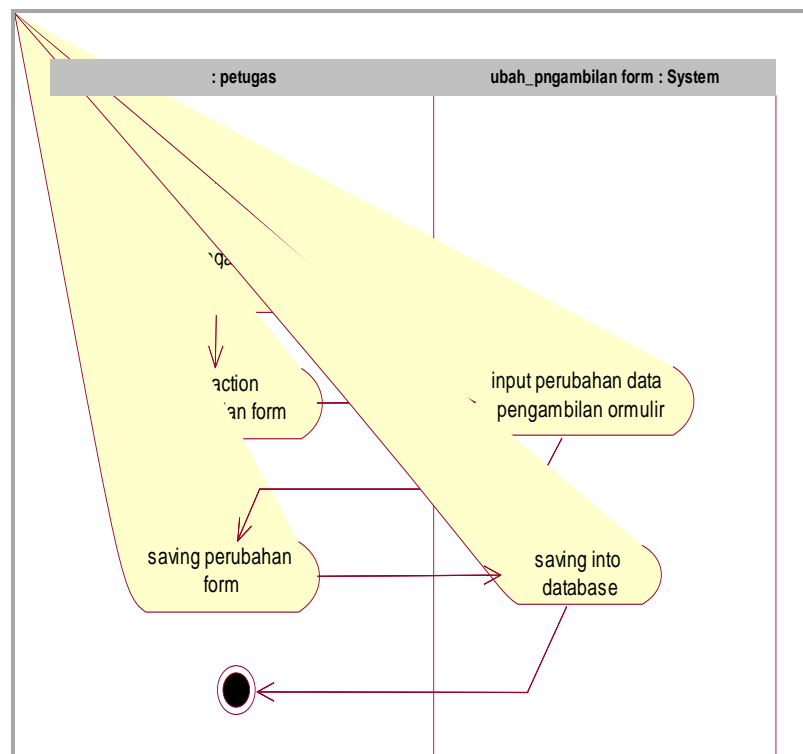
terjadi mulai dari pemilihan menu, input data calon pengambil tes sampai pada penyimpanan data (lihat Gambar 3.6)



Gambar 3.6. Activity Diagram Tambah Pengambilan Formulir

b. Ubah Formulir

Adalah bagaimana proses seorang petugas melakukan edit data atau ubah data dari data atau beberapa data yang sudah pernah disimpan petugas pada system. Aktivitas yang dilakukan mulai dari petugas melakukan *open action* pengambilan formulir, pengubahan data pengambilan formulir sehingga system menyimpan perubahan data di dalam database (lihat Gambar 3.7).



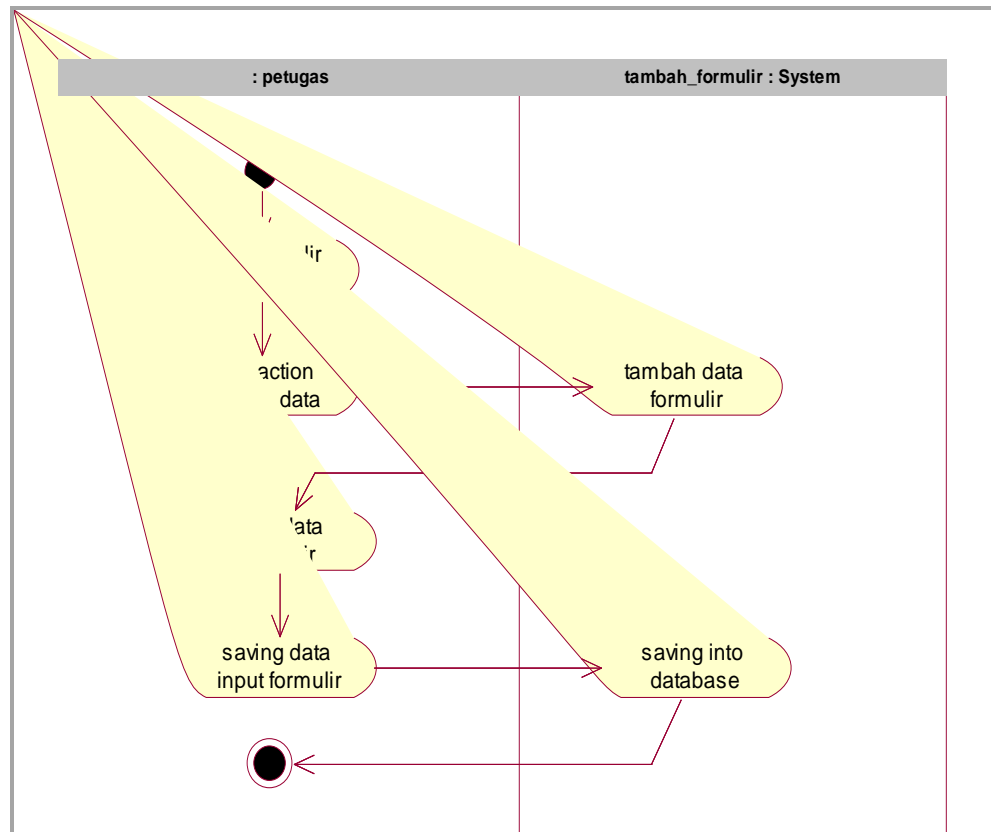
Gambar 3.7. Activity Diagram Ubah Pengambilan Formulir

2) Formulir Pendaftaran

Proses yang ada setelah calon siswa menyerahkan data-data yang berupa formulir pendaftaran, sehingga petugas dapat memasukkan data-data tersebut kedalam *database* melalui *interface system*. Terdapat dua proses yang mungkin dapat dilakukan oleh user/petugas, yang keduanya akan dijabarkan pada gambar dan penjelasan berikut ini.

a. Input Formulir Pendaftaran

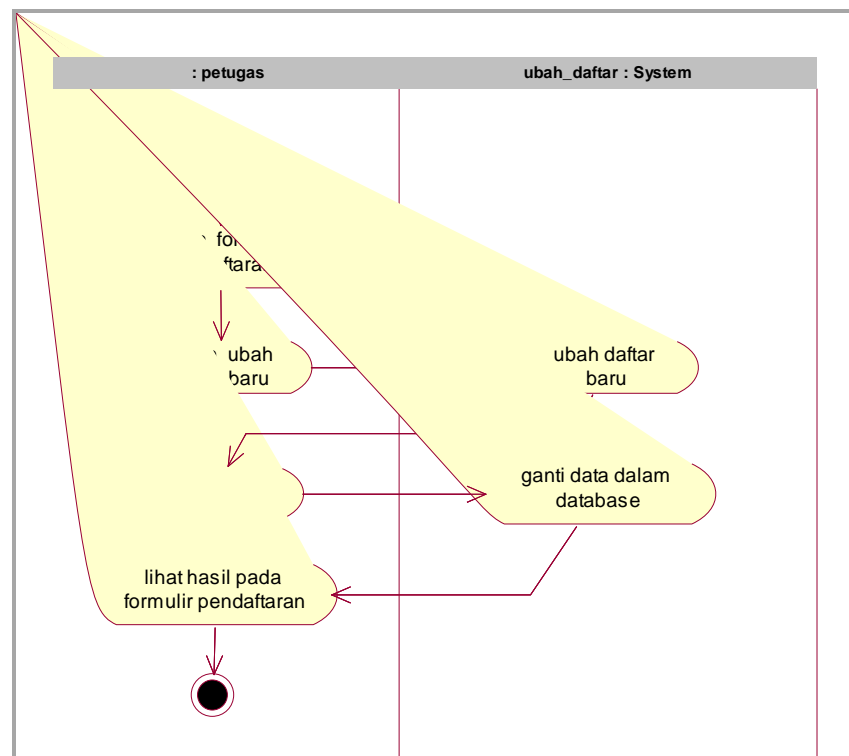
Adalah memasukkan data dari formulir yang telah diserahkan calon siswa kepada petugas. Untuk lebih jelasnya dapat dilihat pada Gambar 3.8.



Gambar 3.8. *Activity Diagram Input Formulir Pendaftaran*

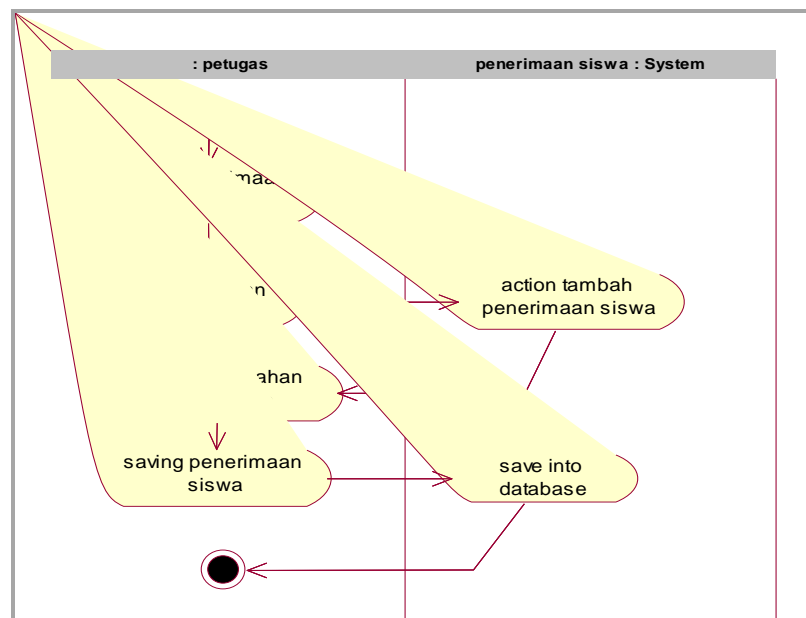
b. Ubah Formulir Pendaftaran

Adalah proses yang terjadi setelah proses input yang dilakukan saat dimana seorang user ingin melakukan perubahan data tersebut dengan membuka *action* ubah data sesuai dengan data yang benar. Seperti pada yang terlihat pada Gambar 3.9.



Gambar 3.9. Activity Diagram Ubah Data Formulir Pendaftaran

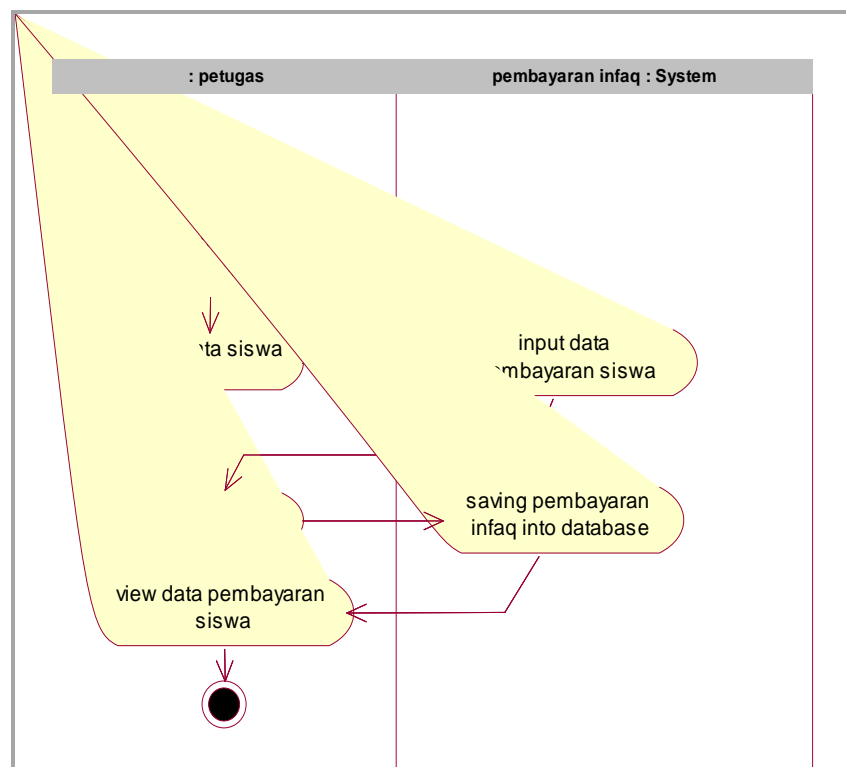
3) Penerimaan Siswa



Gambar 3.10. Activity Diagram Penerimaan Siswa

Proses yang dilakukan adalah ketika user hendak memasukkan data calon siswa yang telah resmi menjadi siswa pada sekolah setelah dilakukan tes masuk menjadi siswa Khadijah 2 dan calon siswa tersebut dinyatakan lulus. Activity diagram tersebut dapat dilihat pada Gambar 3.10.

4) Pembayaran Infaq

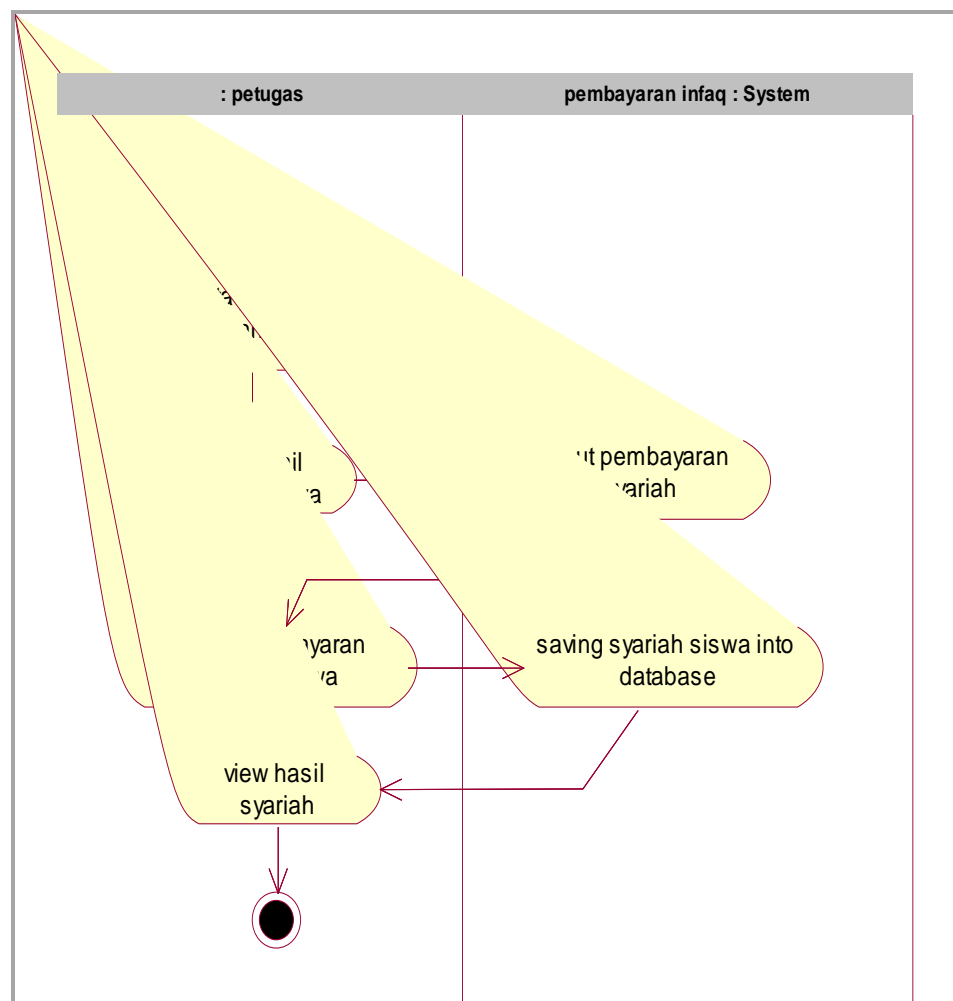


Gambar 3.11. Activity Diagram Pembayaran Infaq

Seperti yang tampak pada Gambar 3.11 tersebut adalah menjelaskan tentang beberapa aktivitas yang terjadi pada saat user hendak mengisi daftar pembayaran infaq setiap siswa yang telah melakukan pembayaran infaq.

5) Pembayaran Syariah

Seperti pada pada pembayaran yang lain, disini pula dilakukan input pembayaran siswa.



Gambar 3.12. Activity Diagram Pembayaran Syahriah

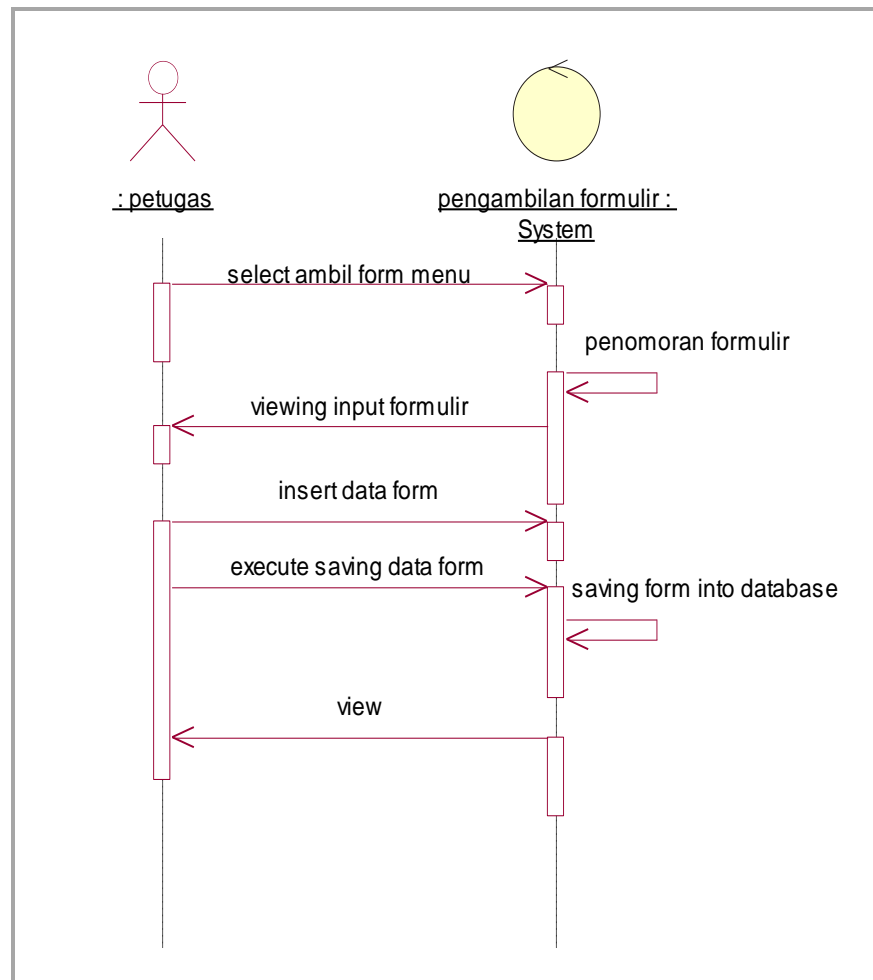
Beberapa aktifitas yang terjadi ketika user akan memasukkan data tentang pembayaran syariah yang telah dilakukan oleh siswa.

3.4.3 Sequence Diagram

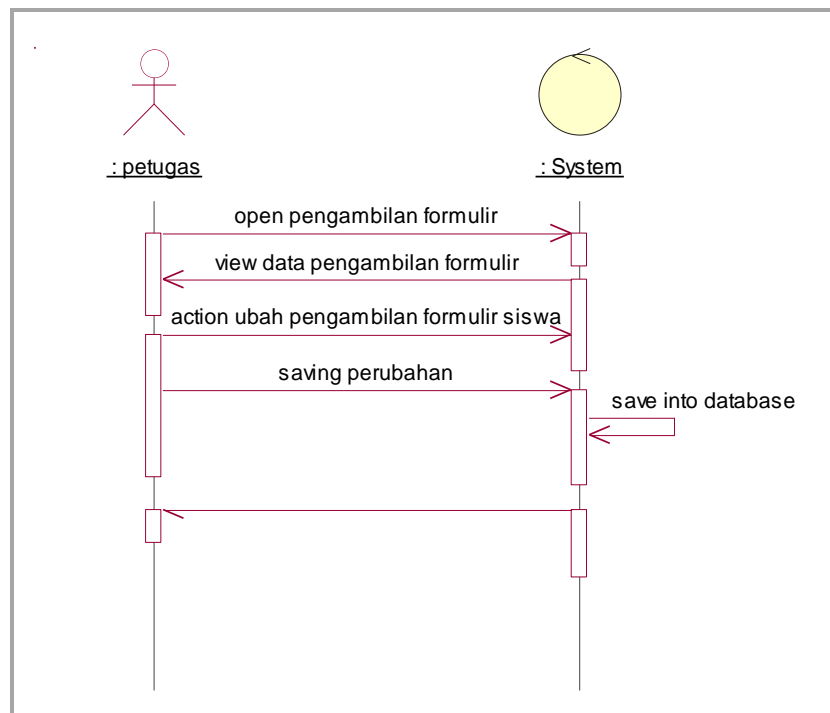
Telah dijelaskan sebelumnya, bahwa setelah selesai dibuat *Activity diagram* proses selanjutnya adalah membuat *sequence diagram* untuk proses pendaftaran siswa dan pembayaran, untuk lebih jelasnya akan dijelaskan pada beberapa gambar dibawah ini.

1) Pengambilan Formulir

Dalam hal ini proses yang terjadi tidak jauh beda dengan penjelasan sebelumnya yaitu pada Activity Diagram. Untuk lebih jelasnya dapat dilihat pada Gambar 3.13 (penambahan data Pengambilan Formulir) dan Gambar 3.14 (pengubahan data Pengambilan Formulir).

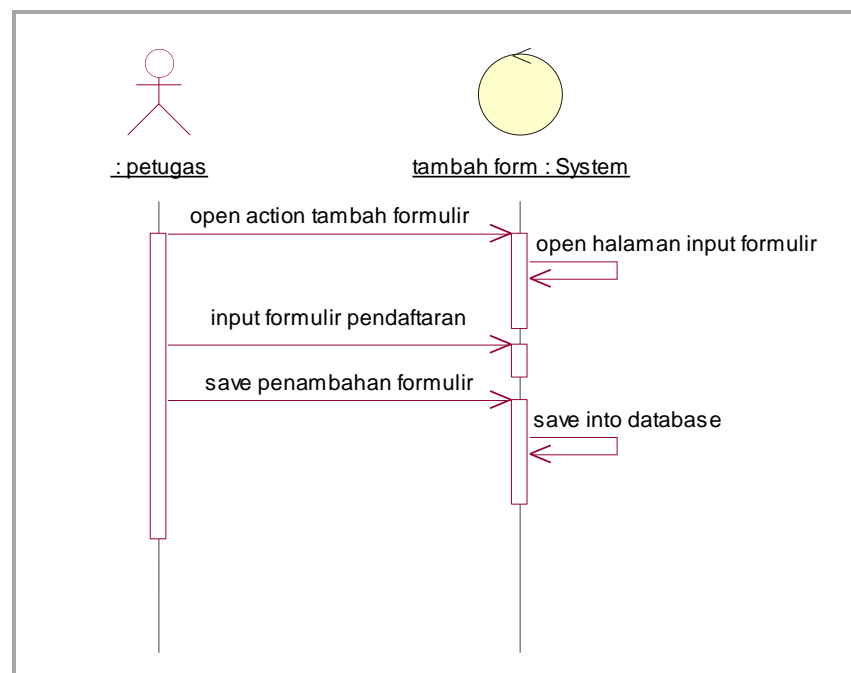


Gambar 3.13. *Sequence Diagram* Tambah Pengambilan Formulir

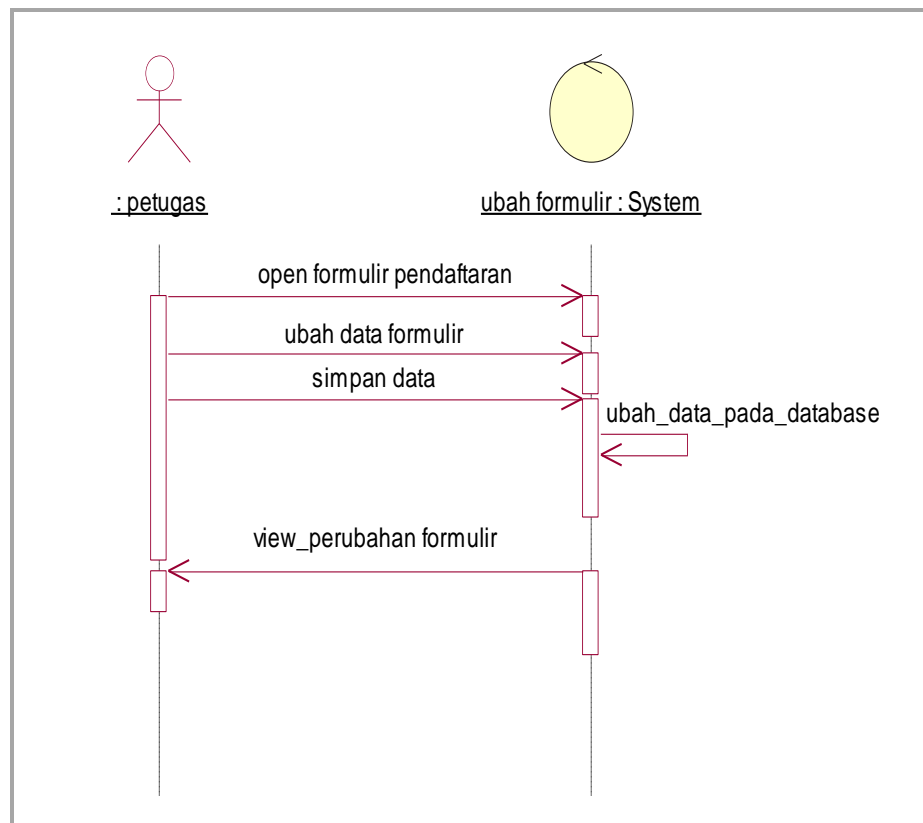


Gambar 3.14. *Sequence Diagram* Ubah Pengambilan Formulir

2) Formulir Pendaftaran



Gambar 3.15. *Sequence Diagram* Tambah Formulir Pendaftaran

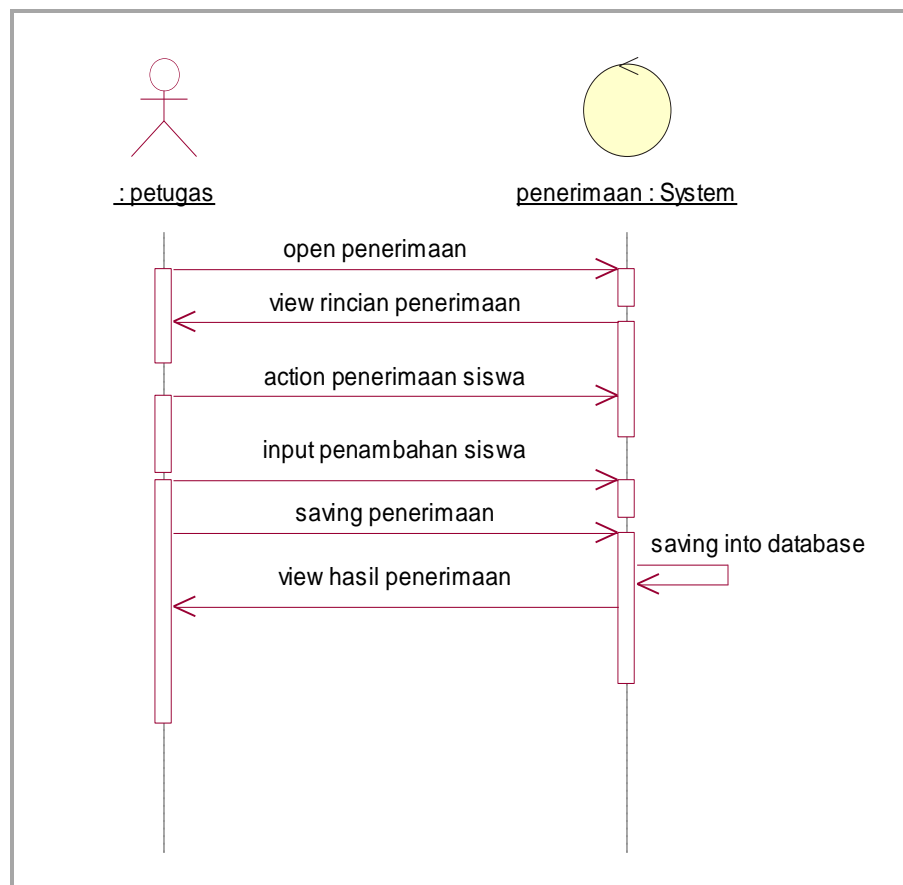


Gambar 3.16. *Sequence Diagram* Ubah Formulir Pendaftaran

Seperti yang terlihat pada Gambar 3.15 dan Gambar 3.16, gambar tersebut menjelaskan mengenai *Sequence Diagram* Formulir Pendaftaran yang tidak lain adalah bagian dari *Activity Diagram* Formulir Pendaftaran.

3) Penerimaan Siswa

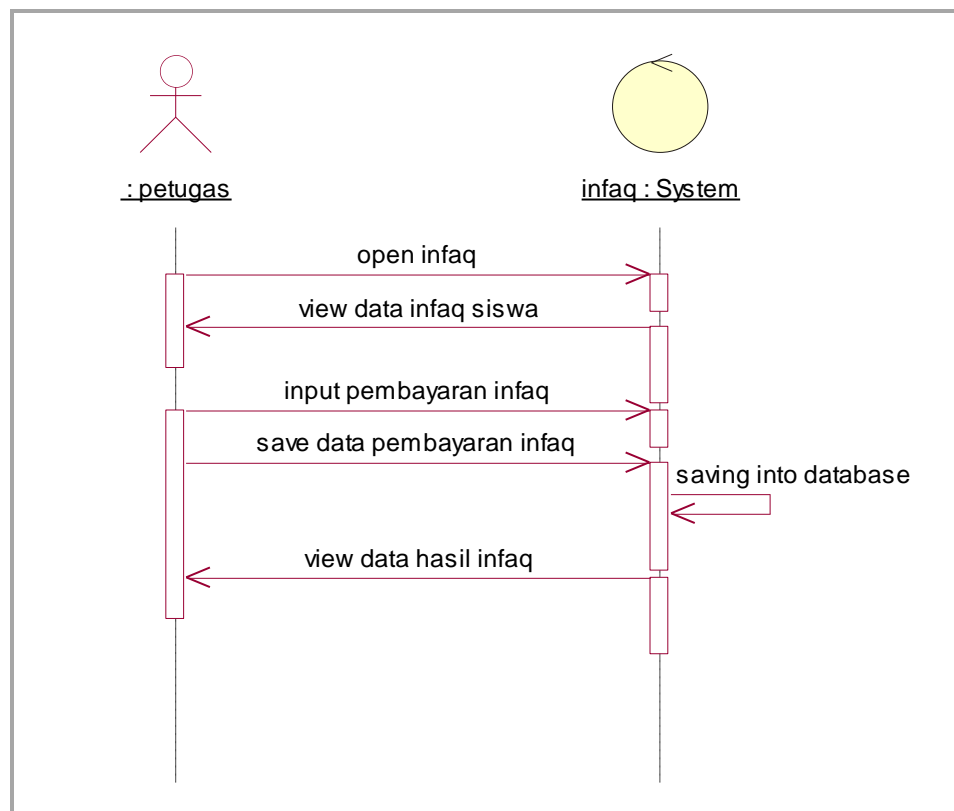
Dalam hal ini, *Sequence Diagram* Penerimaan Siswa ini adalah masih merupakan bagian dari *Activity Diagram* mengenai Penerimaan Siswa. *Sequence Diagram* yang ada di sini telah dijelaskan secara gamblang seperti pada Gambar 3.17.



Gambar 3.17. *Sequential Diagram* Penerimaan Siswa

4) Pembayaran Infaq

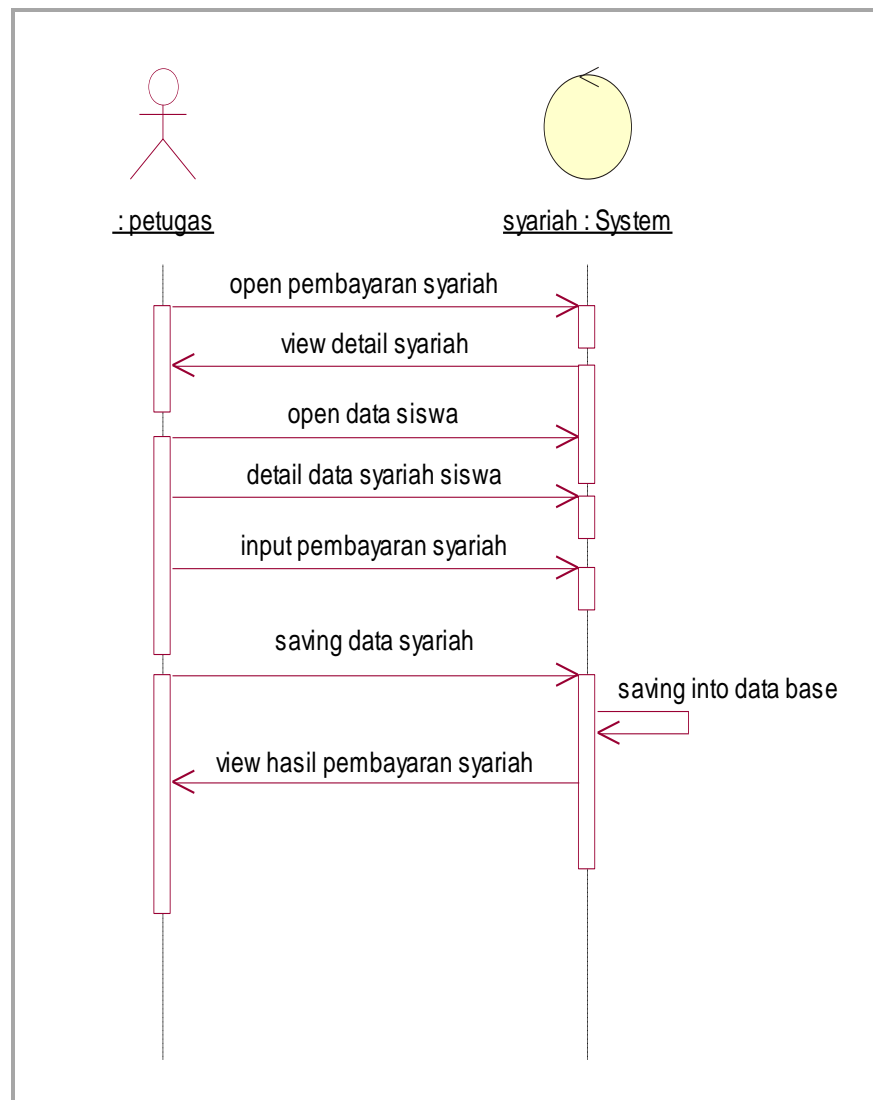
Dalam hal ini, *Sequence Diagram* Pembayaran Infaq ini adalah masih merupakan bagian dari *Activity Diagram* mengenai Pembayaran Infaq. *Sequence Diagram* yang ada di sini telah dijelaskan secara gamblang seperti pada Gambar 3.18.



Gambar 3.18. *Sequence Diagram* Pembayaran Infaq

5) Pembayaran Syariah

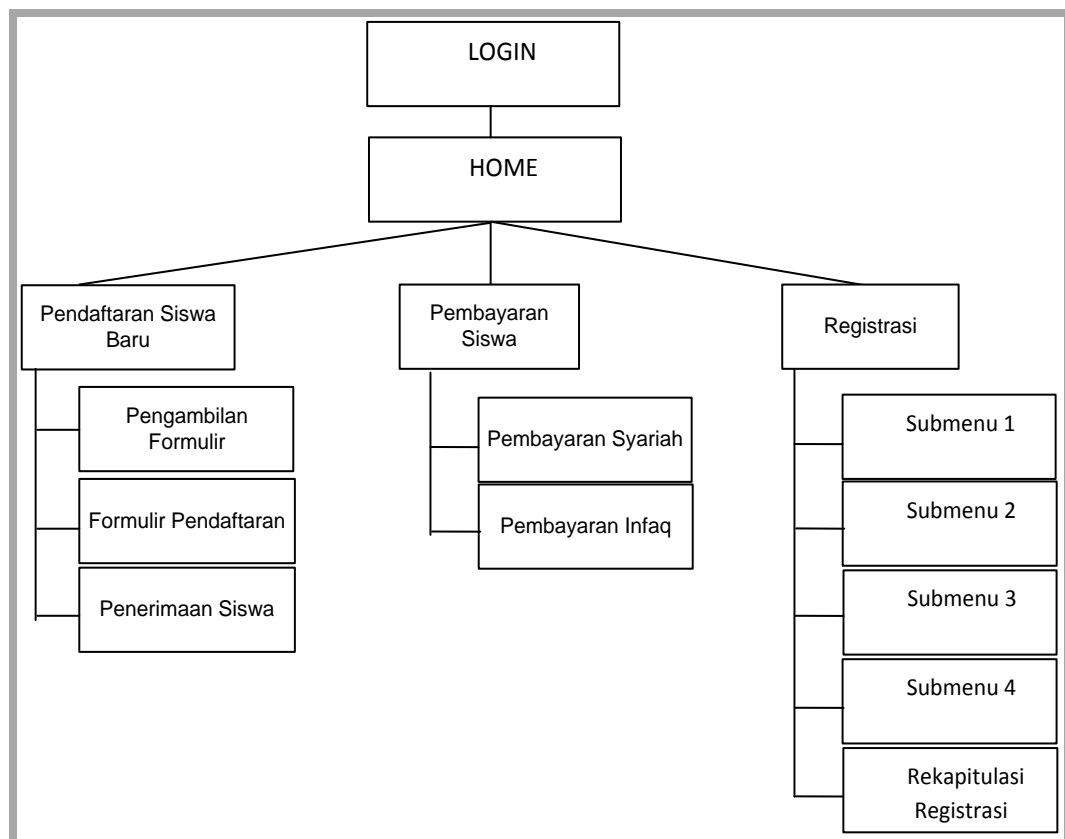
Dalam hal ini, *Sequence Diagram* Pembayaran Syariah ini adalah masih merupakan bagian dari *Activity Diagram* mengenai Pembayaran Infaq. *Sequence Diagram* yang ada di sini telah dijelaskan seperti pada Gambar 3.19 baik mengenai actor yang terlibat dalam system dan proses-proses yang ada pada system.



Gambar 3.19. *Sequence Diagram* Pembayaran Syariah

3.5 Perancangan Sistem Menu Program

Dalam merancang program *website* dibuat agar terkesan menarik dan memudahkan user untuk mengakses, maka penulis membuat sistem menu seperti yang terdapat pada gambar 3.20.



Gambar 3.20. Rancangan Sistem Informasi.

Untuk penjelasan sistem menu tersebut akan diuraikan pada deskripsi-deskripsi berikut dibawah ini :

a. Login

Bagian ini menjadi halaman pertama yang akan tampil ketika situs *website* ini dibuka, hal ini dimaksudkan untuk *user* yang akan mengakses system informasi ini. Ibarat pintu rumah untuk dapat melakukan masuk harus memiliki sebuah kunci yang sesuai sama halnya halaman login harus mempunyai kata kunci (*username, password*) yang sesuai agar dapat mengakses sistem tersebut. *User* yang belum teregistrasi akan disediakan form khusus untuk melakukan registrasi.

The diagram shows a rectangular login form with a black border. Inside the form, the text 'Username' is positioned above a rectangular input field. Below this, the text 'Password' is positioned above another rectangular input field. At the bottom of the form, there is a rectangular button labeled 'login'. Three horizontal lines with bold numbers at their right ends point to these elements: line '1' points to the Username input field, line '2' points to the Password input field, and line '3' points to the login button.

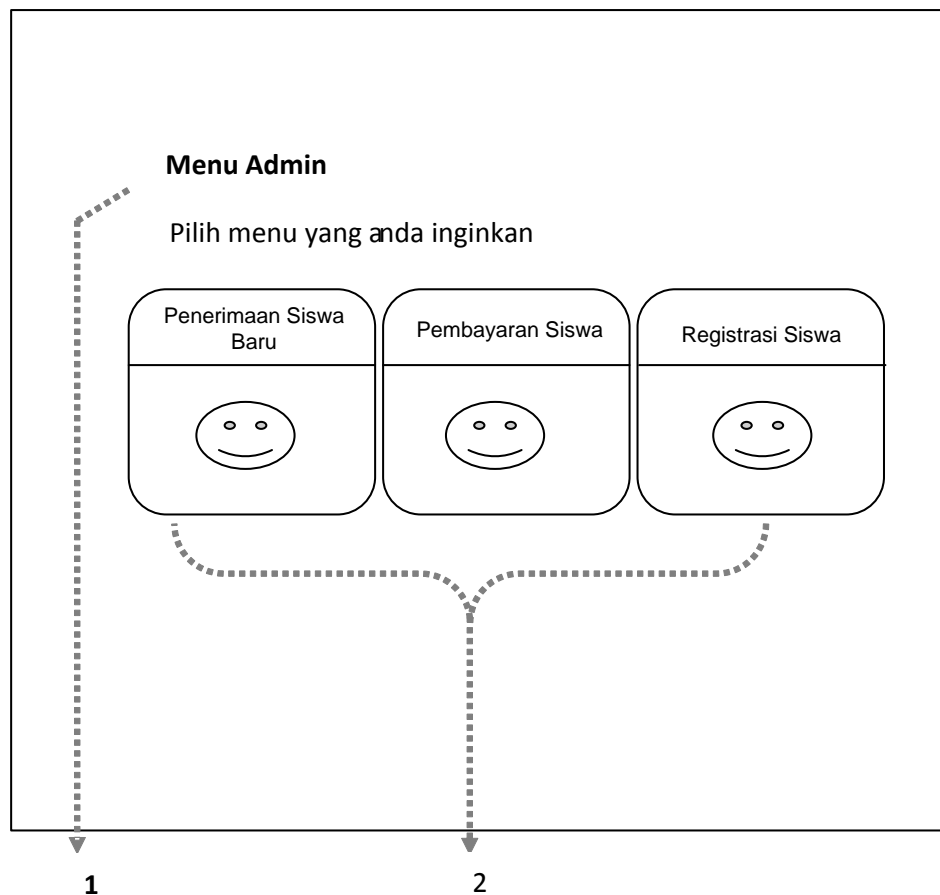
Gambar 3.21. Rancangan Form *Login* untuk Autentikasi User

Keterangan gambar :

1. *User name* : untuk *identifikasi* nama hak akses user
2. *Password* : kode keamanan user
3. *Login* : *key button* untuk eksekusi no.1 dan no.2

b. Menu Admin

Menu ini dapat diakses oleh user jika proses login berjalan dengan benar dalam artian bahwa username dan password yang dimasukkan sesuai dengan yang ada pada database, menu ini adalah menu yang pertama kali yang dilihat oleh user setelah melakukan setelah halaman login, menu ini berfungsi sebagai menu utama, dan secara garis besar pilihan yang tersedia adalah *pendaftaran siswa*, *pembayaran siswa* dan *registrasi siswa*.



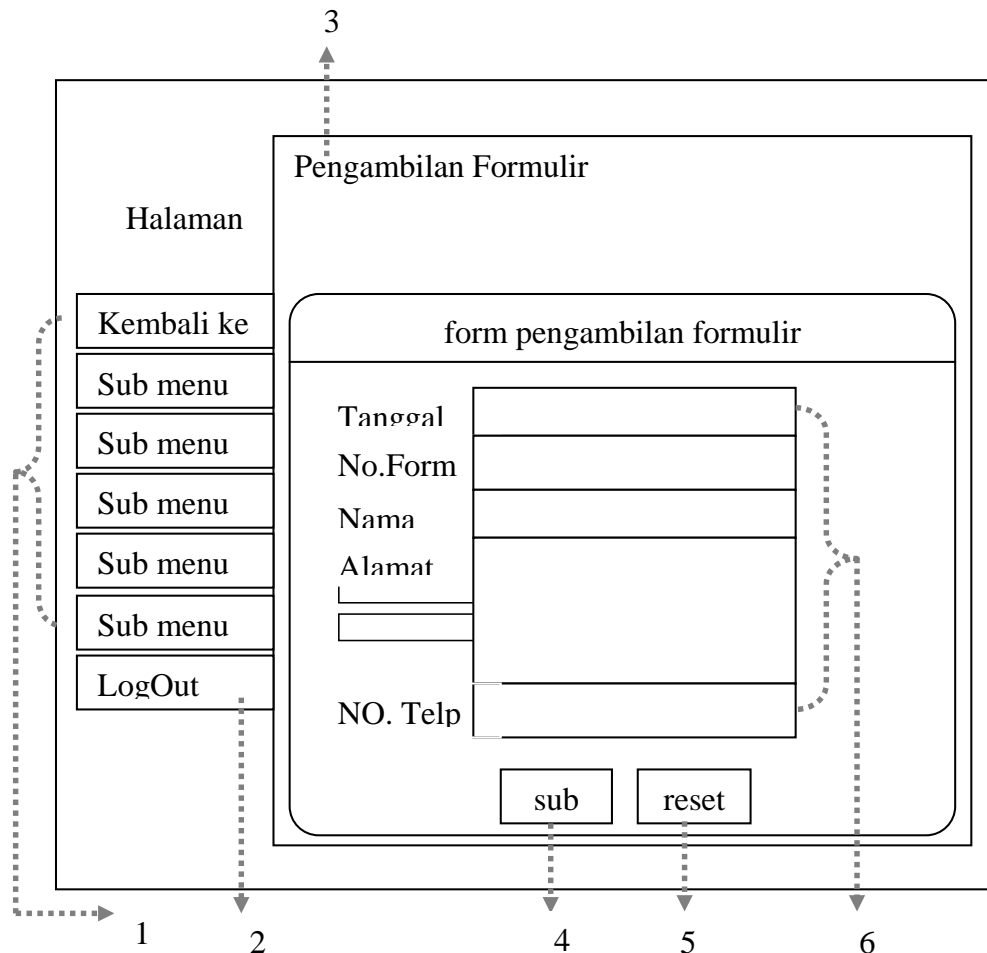
Gambar 3.22. Rancangan Menu Admin

Seperti yang telah terlihat pada gambar tersebut judul sebagai header dan kotak-kotak *button* sebagai pilihan langkah selanjutnya.

Untuk lebih memudahkan meamahami menu admin ini bisa dilanjutkan sebagaimana dalam keterangan berikut :

- 1) Header , berfungsi unyuk mengidentifikasi judul pada tampilan form.
- 2) Menu , adalah rancangan tampilan menu pada form dan yang kita lihat dalam gambar disitu terdapat nama menu yang disertai gambar icon yang berbeda pada tiap menu agar form terlihat lebih menarik.

Kemudian setelah tampilan sub-menu pada menu admin, user dapat memilih salah satu dari menu sesuai dengan kebutuhannya pada sistem. Tampak pada gambar 3.23 dibawah ini.



Gambar 3.23. Rancangan Tampilan Form

Keterangan dan penjelasan detail gambar :

- a. Sub-menu yang ada.
- b. *Button* sub menu *logout* ,apabila akses sistem dapat menuser yang ingin meninggalkan atau keluar dari sistem tersebut dan untuk *kembali ke* digunakan untuk menuju pada menu sebelumnya.

- c. *Header* halaman yang dicontohan disini adalah sub-menu pengambilan formulir
- d. *Button submit*, yaitu eksekusi inputan untuk memasukkan data pada *database*.
- e. *Button reset* menghilangkan data pada kotak-kotak inputan.
- f. Kotak-kotak inputan, disini adalah tempat dimana *user* dapat melakukan aktifitasnya untuk memasukkan data dalam *database*, misalnya nama, alamat, tanggal, dst. Seperti yang ada pada gambar.

BAB IV

IMPLEMENTASI SISTEM

Pada bab IV ini akan dibahas mengenai implementasi dari rancangan sistem yang telah dibuat pada bab III. Bagian implementasi sistem kali ini meliputi: lingkungan implementasi, implementasi proses, dan implementasi antarmuka.

4.1 Lingkungan Implementasi

Pada bagian ini akan dijelaskan mengenai perangkat keras dan perangkat lunak yang digunakan pada implementasi sistem ini.

Perangkat Keras :

- a. Notebook dengan prosessor Intel Core 2 Duo Inside
- b. Memori 2 GB
- c. Harddisk 160 GB

Perangkat Lunak

- a. Sistem Operasi Windows 7
- b. Macromedia Dreamweaver 8 untuk mendesain aplikasi ini
- c. PHP AJAX
- d. Apache Triad
- e. Mozilla Firefox (untuk menjalankan aplikasi di localhost maupun di web)
- f. Power Designer 11
- g. Microsoft Office 2010

4.2 Implementasi Data

Pada tahap ini akan dibahas mengenai implementasi data dari perancangan data yang telah dibahas sebelumnya. Dari hasil *generate database* didapat script untuk membuat tabel – tabel dalam *database* yang akan dibuat. Proses pembuatan basis data diawali dengan perancangan CDM (Conceptual Data Model) pada Power Designer 12 yang kemudian di-*generate* menjadi PDM (Physical Data Model). Adapun tabel-tabel yang digunakan dalam aplikasi ini adalah sebagai berikut,

1) Tabel Ambil Formulir

Gambar 4.1 merupakan gambar tabel ambil formulir dimana *id_ambil* menjadi *primary_key*. Tabel *ambil_formulir* ini mempunyai delapan field dan di tabel *ambil_formulir* ini mempunyai satu *foreign_key* yaitu *id_tahun_ajaran* yang menjadi *foreign_key* dari *primary_key* tabel *tahun_ajaran*. Karena di tabel *ambil_formulir* ini akan menjadi awal dari proses pembentukan tahun ajaran baru.

```

/*=====*/
/* Table: AMBIL_FORMULIR */
/*=====*/
create table AMBIL_FORMULIR
(
    ID_AMBIL                int(11) not null auto_increment,
    ID_TAHUN_AJARAN         int(11),
    TGL_AMBIL               date not null,
    NO_FORM                 int not null,
    NAMA_PENGAMBIL          varchar(50) not null,
    ALAMAT_PENGAMBIL        varchar(100) not null,
    TELP_PENGAMBIL          varchar(50) not null,
    ASAL_SEKOLAH            varchar(100) not null,
    primary key (ID_AMBIL)
);

```

Gambar 4.1. Tabel Ambil Formulir

2) Tabel Daftar Ulang

Gambar 4.2 merupakan gambar tabel daftar ulang dimana yang menjadi `primary_key` dari tabel ini adalah `id_reg`. Tabel `daftar_ulang` tersebut memiliki lima field, dan terdapat dua `foreign_key` yang berelasi pada tabel ini yaitu `id_siswa` dari tabel `siswa` dan `id_tahun_ajaran` dari tabel `tahun_ajaran`. Masuknya `id_siswa` dan `id_tahun_ajaran` dalam tabel `daftar_ulang` merupakan relasi yang saling terkait, karena tabel ini membutuhkan `id_siswa` dan `id_tahun_ajaran` untuk dapat diproses selanjutnya.

```

/*=====*/
/* Table: DAFTAR_ULANG */
/*=====*/
create table DAFTAR_ULANG
(
    ID_REG                int(11) not null auto_increment,
    ID_SISWA              int(11),
    ID_TAHUN_AJARAN       int(11),
    TGL_REG               date not null,
    DARI_KELAS            varchar(5) not null,
    KE_KELAS              varchar(5) not null,
    STATUS_REG            int not null,
    primary key (ID_REG)
);

```

Gambar 4.2. Tabel Daftar Ulang

3) Tabel Formulir

Gambar 4.3 merupakan gambar tabel formulir dimana pada tabel ini terdapat banyak field yang menjadi bagian dari tabel formulir. Karena tabel ini merupakan data dari calon siswa baru yang telah mengambil formulir dan mengisi formulir tersebut. `Id_form` merupakan `primary_key` dari tabel formulir dan `id_ambil` masuk sebagai `foreign_key` dari tabel `ambil_formulir`.

```

/*=====*/
/*Table:FORMULIR                                     */
/*=====*/
create table FORMULIR
(
    ID_FORM                int(10) not null auto_increment,
    ID_AMBIL                int(11),
    NAMA_PENDAFTAR          varchar(100) not null,
    STATUS_PENDAFTAR        int not null,
    ALAMAT_PENDAFTAR        text not null,
    TELP_PENDAFTAR          varchar(50) not null,
    NAMA LENGKAP            varchar(100) not null,
    NAMA_PANGGILAN          varchar(50) not null,
    JENKEL                  varchar(2) not null,
    TGL_LHR                 date not null,
    TMPT_LHR                varchar(50) not null,
    AGAMA                   varchar(10) not null,
    KEWARGANEGARAAN         int not null,
    ANAK_KE                 int not null,
    KANDUNG                 int not null,
    TIRI                    int not null,
    ANGKAT                  int not null,
    BAHASA                  varchar(40) not null,
    GOLONGAN_DARAH          varchar(2) not null,
    ALAMAT_SISWA            text not null,
    TELP_SISWA              varchar(15) not null,
    TINGGAL_PADA            text not null,
    JARAK_SEKOLAH           varchar(10) not null,
    PHOTO                   longblob not null,
    NAMA_AYAH               varchar(100) not null,
    NAMA_IBU                varchar(100) not null,
    PENDIDIKAN_AYAH         varchar(100) not null,
    PENDIDIKAN_IBU          varchar(100) not null,
    GURU_AYAH               int not null,
    PEKERJAAN_AYAH          varchar(100) not null,
    GURU_IBU                int not null,
    PEKERJAAN_IBU           varchar(100) not null,
    NAMA_WM                 varchar(100) not null,
    HUB_KELUARGA            varchar(100) not null,
    PENDIDIKAN_WM           varchar(100) not null,
    PEKERJAAN_WM            varchar(100) not null,
    ASAL_MURID              varchar(100) not null,
    NAMA_TK                 varchar(100) not null,
    TGL_NO_STTB             varchar(100) not null,
    ASAL_SEKOLAH            varchar(100) not null,
    DARI_TINGKAT            varchar(50) not null,
    DITERIMA_TGL            date not null,
    TAHUN1                  date not null,
    BERAT_BADAN1            varchar(2) not null,
    TINGGI_BADAN1           varchar(3) not null,
    PENYAKIT1              varchar(50) not null,
    KELAINAN1              varchar(50) not null,
    TAHUN2                  date not null,
    BERAT_BADAN2            varchar(2) not null,

```


TINGGI_BADAN2	varchar(3) not null,
PENYAKIT2	varchar(50) not null,
KELAINAN2	varchar(50) not null,
TAHUN3	date not null,
BERAT_BADAN3	varchar(2) not null,
TINGGI_BADAN3	varchar(3) not null,
PENYAKIT3	varchar(50) not null,
KELAINAN3	varchar(50) not null,
TAHUN4	date not null,
BERAT_BADAN4	varchar(2) not null,
TINGGI_BADAN4	varchar(3) not null,
PENYAKIT4	varchar(50) not null,
KELAINAN4	varchar(50) not null,
TAHUN5	date not null,
BERAT_BADAN5	varchar(2) not null,
TINGGI_BADAN5	varchar(3) not null,
PENYAKIT5	varchar(50) not null,
KELAINAN5	varchar(50) not null,
TAHUN6	date not null,
BERAT_BADAN6	varchar(2) not null,
TINGGI_BADAN6	varchar(3) not null,
PENYAKIT6	varchar(50) not null,
KELAINAN6	varchar(50) not null,
JNS_BEASISWA	varchar(100) not null,
TAHUN_TAMAT	date not null,
TAMAT_NO_STTB	varchar(10) not null,
MELANJUTKAN_SEKOLAH	varchar(100) not null,
PINDAH_SEKOLAH	varchar(100) not null,
DADI_TINGKAT2	varchar(5) not null,
KE_SEKOLAH	varchar(100) not null,
TANGGAL_PINDAH	date not null,
KELUAR_SEKOLAH	varchar(100) not null,
TANGGAL_KELUAR	date not null,
ALASAN_KELUAR	text not null,
CATATAN	text not null,
primary key (ID_FORM)	
);	

Gambar 4.3. Tabel Formulir

4) Tabel Infaq

Gambar 4.4 merupakan gambar tabel infaq dimana dalam tabel tersebut terdapat satu foreign_key yaitu id_siswa dan id_infaq menjadi primary_key. Tabel ini membutuhkan id_siswa untuk dapat melakukan proses pembayaran infaq yang dibebankan kepada siswa baru yang telah diterima sebagai siswa SD Khadijah 2 Surabaya.

```

/*=====*/
/* Table: INFAQ */
/*=====*/
create table INFAQ
(
    ID_INFAQ          int(11) not null auto_increment,
    ID_SISWA          int(11),
    TGL_BAYAR_INFAQ   date not null,
    JUMLAH_DIBAYAR    varchar(20) not null,
    primary key (ID_INFAQ)
);

```

Gambar 4.4. Tabel Infaq

5) Tabel Penerimaan

Gambar 4.5 merupakan gambar tabel penerimaan dimana id_penerimaan menjadi primary_key dan id_form masuk ke dalam tabel penerimaan sebagai foreign_key.

```

/*=====*/
/* Table: PENERIMAAN */
/*=====*/
create table PENERIMAAN
(
    ID_PENERIMAAN      int not null auto_increment,
    ID_FORM             int(10),
    TGL_PENERIMAAN     date not null,
    GELOMBANG          int not null,
    DITERIMA_DI_KELAS  varchar(5) not null,
    INFAQ              varchar(20) not null,
    STATUS_INFAQ       int not null,
    STATUS_PENERIMAAN  int not null,
    primary key (ID_PENERIMAAN)
);

```

Gambar 4.5. Tabel Penerimaan

6) Tabel Siswa

Gambar 4.6 merupakan gambar tabel siswa dimana primary_key tabel tersebut adalah id_siswa dan id_penerimaan menjadi foreign_key dalam tabel siswa.

```

/*=====*/
/* Table: SISWA */
/*=====*/
create table SISWA
(
    ID_SISWA          int(11) not null auto_increment,
    ID_PENERIMAAN     int,
    NIS               varchar(30) not null,
    NAMA_SISWA        varchar(50) not null,
    KELAS_SISWA        varchar(3) not null,
    primary key (ID_SISWA)
);

```

Gambar 4.6. Tabel Siswa

7) Tabel SPP / Syahriah

Gambar 4.7 merupakan gambar tabel spp / syahriah dimana id_spp sebagai primary_key, sedangkan yang menjadi foreign_key ada dua yaitu id_tahun_ajaran dan id_siswa. Tabel ini memerlukan dua primary_key dari tabel siswa dan tabel tahun_ajaran untuk masuk kedalam foreign_key karena id_tahun_ajaran dan id_siswa berperan penting dalam proses pembayaran spp.

```

/*=====*/
/* Table: SPP */
/*=====*/
create table SPP
(
    ID_SPP            int not null auto_increment,
    ID_TAHUN_AJARAN   int(11),
    ID_SISWA           int(11),
    BLN_SPP1           int not null,
    BLN_SPP2           int not null,
    BLN_SPP3           int not null,
    BLN_SPP4           int not null,
    BLN_SPP5           int not null,
    BLN_SPP6           int not null,
    BLN_SPP7           int not null,
    BLN_SPP8           int not null,
    BLN_SPP9           int not null,
    BLN_SPP10          int not null,
    BLN_SPP11          int not null,
    BLN_SPP12          int not null,
    TGL_SPP1           date not null,
    TGL_SPP2           date not null,
    TGL_SPP3           date not null,
    TGL_SPP4           date not null,

```

```

TGL_SPP5          date not null,
TGL_SPP6          date not null,
TGL_SPP7          date not null,
TGL_SPP8          date not null,
TGL_SPP9          date not null,
TGL_SPP10         date not null,
TGL_SPP11         date not null,
TGL_SPP12         date not null,
primary key (ID_SPP)
);

```

Gambar 4.7. Tabel SPP / Syahriah

8) Tabel Tahun Ajaran

Gambar 4.8 merupakan tabel tahun ajaran yang mana di dalam tabel ini hanya terdapat dua field. Id_tahun_ajaran sebagai primary_key dan di dalam tabel ini tidak terdapat foreign_key.

```

/*=====*/
/* Table: TAHUN_AJARAN */
/*=====*/
create table TAHUN_AJARAN
(
    ID_TAHUN_AJARAN    int(11) not null auto_increment,
    TAHUN_AJARAN       varchar(10) not null,
    primary key (ID_TAHUN_AJARAN)
);

```

Gambar 4.8. Tabel Tahun Ajaran

9) Tabel User

Gambar 4.9 merupakan gambar tabel user yang mana tabel ini seperti tabel tahun ajaran yang dalam tabel ini tidak ditemukan foreign_key. Tabel user memiliki tiga field yaitu username, password dan id_user. Primary_key dari tabel user adalah id_user.

```

/*=====*/
/* Table: USER */
/*=====*/
create table USER
(
    ID_USER          int(11) not null auto_increment,
    USERNAME         varchar(30) not null,
    PASSWORD         varchar(100) not null,
    primary key (ID_USER)
);

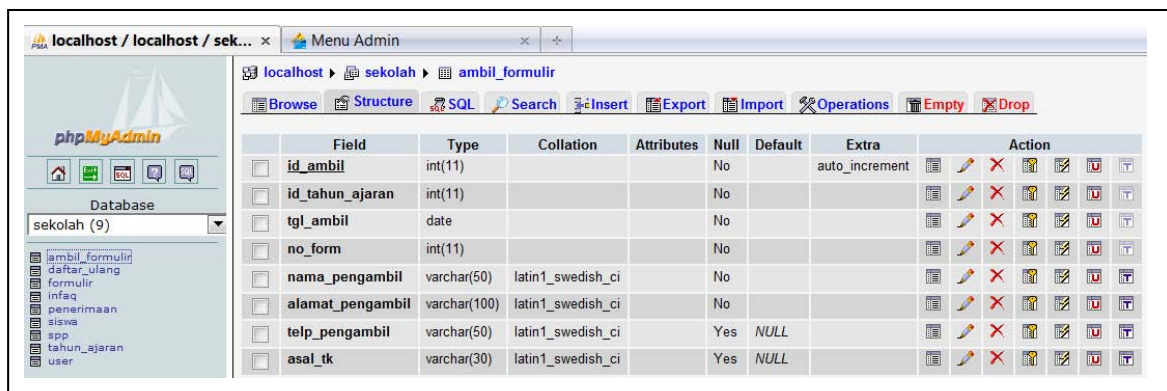
```

Gambar 4.9. Tabel User

Setelah tabel-tabel tersebut selesai di generate ke dalam database, langkah selanjutnya adalah memasukkan hasil generate database tersebut kedalam database PHPMYAdmin seperti gambar di bawah ini :

1) Database Ambil Formulir

Gambar 4.10 merupakan database dari ambil formulir. Database ini menyimpan data pengambil formulir yang membeli formulir PSB (Pendaftaran Siswa Baru). Id_ambil merupakan primary_key dari database ambil formulir ini, sedangkan id_tahun_ajaran masuk sebagai foreign_key dimana database ambil formulir membutuhkan id_tahun_ajaran untuk memulai proses berlangsungnya tahun ajaran baru.



Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id_ambil	int(11)			No		auto_increment	[Icons]
<input type="checkbox"/> id_tahun_ajaran	int(11)			No			[Icons]
<input type="checkbox"/> tgl_ambil	date			No			[Icons]
<input type="checkbox"/> no_form	int(11)			No			[Icons]
<input type="checkbox"/> nama_pengambil	varchar(50)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> alamat_pengambil	varchar(100)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> telp_pengambil	varchar(50)	latin1_swedish_ci		Yes	NULL		[Icons]
<input type="checkbox"/> asal_tk	varchar(30)	latin1_swedish_ci		Yes	NULL		[Icons]

Gambar 4.10 Database Ambil Formulir

2) Database Daftar Ulang

Gambar 4.11 merupakan database dari daftar ulang. Database ini menyimpan data siswa yang melakukan daftar ulang / registrasi. Database daftar ulang ini hanya bs diproses satu tahun sekali setiap tahun ajaran baru. Id_reg merupakan primary_key dari database daftar ulang, sedangkan id_tahun_ajaran dan id_siswa masuk sebagai foreign_key.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id_reg	int(11)			No		auto_increment	[Icons]
<input type="checkbox"/> id_tahun_ajaran	int(11)			No			[Icons]
<input type="checkbox"/> id_siswa	int(11)			No			[Icons]
<input type="checkbox"/> tgl_reg	date			No			[Icons]
<input type="checkbox"/> dari_kelas	varchar(5)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> ke_kelas	varchar(5)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> status_reg	int(2)			No			[Icons]

Gambar 4.11 Database Daftar Ulang

3) Database Formulir

Gambar 4.12 merupakan database formulir. Database ini menyimpan data calon siswa yang telah mengisi formulir. Id_ambil merupakan primary_key dari database ambil formulir ini, sedangkan id_tahun_ajaran masuk sebagai foreign_key dimana database ambil formulir membutuhkan id_tahun_ajaran untuk memulai proses berlangsungnya tahun ajaran baru.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id_form	int(10)			No		auto_increment	[Icons]
<input type="checkbox"/> id_ambil	int(11)			No			[Icons]
<input type="checkbox"/> nama_pendaftar	varchar(100)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> status_pendaftar	int(2)			No			[Icons]
<input type="checkbox"/> alamat_pendaftar	longtext	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> telp_pendaftar	varchar(50)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> nama_lengkap	varchar(100)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> nama_panggilan	varchar(50)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> jenkel	varchar(2)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> tgl_lhr	date			No			[Icons]
<input type="checkbox"/> tempat_lhr	varchar(50)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> agama	varchar(10)	latin1_swedish_ci		No			[Icons]
<input type="checkbox"/> kewarganegaraan	int(2)			No			[Icons]
<input type="checkbox"/> anak_ke	int(2)			No			[Icons]

Gambar 4.12 Database Formulir

4) Database Infaq

Gambar 4.13 merupakan database infaq. Database ini menyimpan data pembayaran infaq siswa yang telah diterima dan terbagi menjad dua gelombang. Id_infaq merupakan primary_key dari database infaq ini, sedangkan id_siswa masuk sebagai foreign_key dimana database infaq membutuhkan id_siswa untuk memulai proses berlangsungnya pembayaran infaq.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id_infaq	int(11)			No		auto_increment	[Icons]
<input type="checkbox"/> id_siswa	int(11)			No			[Icons]
<input type="checkbox"/> tgl_bayar_infaq	date			No			[Icons]
<input type="checkbox"/> jumlah_dibayar	varchar(15)	latin1_swedish_ci		No			[Icons]

Gambar 4.13. Database Infaq

5) Database Penerimaan

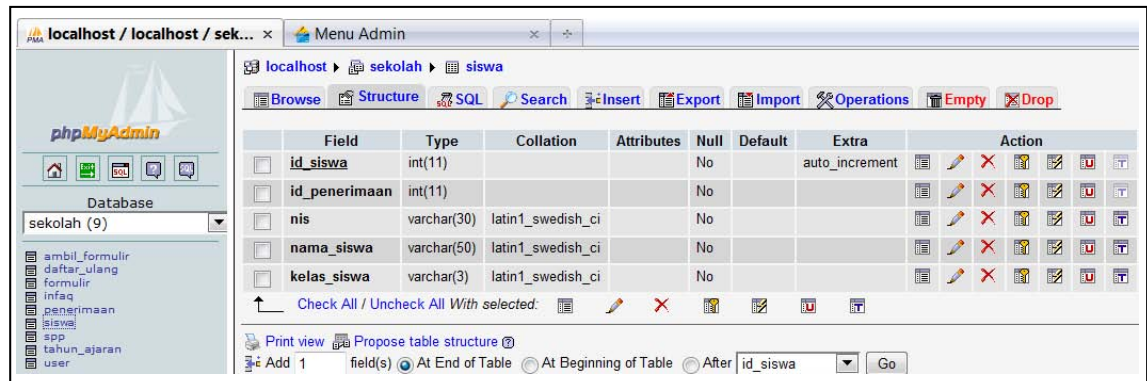
Gambar 4.14 merupakan database penerimaan. Database ini menyimpan data siswa baru yang sudah diterima. Dan pembagian kelas juga disimpan dalam database ini. Id_penerimaan merupakan primary_key dari database penerimaan ini, sedangkan id_form masuk sebagai foreign_key dimana database penerimaan membutuhkan id_form untuk memulai proses berlangsungnya penerimaan siswa.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> id_penerimaan	int(11)			No		auto_increment	
<input type="checkbox"/> id_form	int(11)			No			
<input type="checkbox"/> tgl_penerimaan	date			No			
<input type="checkbox"/> gelombang	int(2)			No			
<input type="checkbox"/> diterima_di_kelas	varchar(5)	latin1_swedish_ci		No			
<input type="checkbox"/> infaq	int(11)			No			
<input type="checkbox"/> status_infaq	int(2)			No			
<input type="checkbox"/> status_penerimaan	int(11)			No	0		

Gambar 4.14. Database Penerimaan

6) Database Siswa

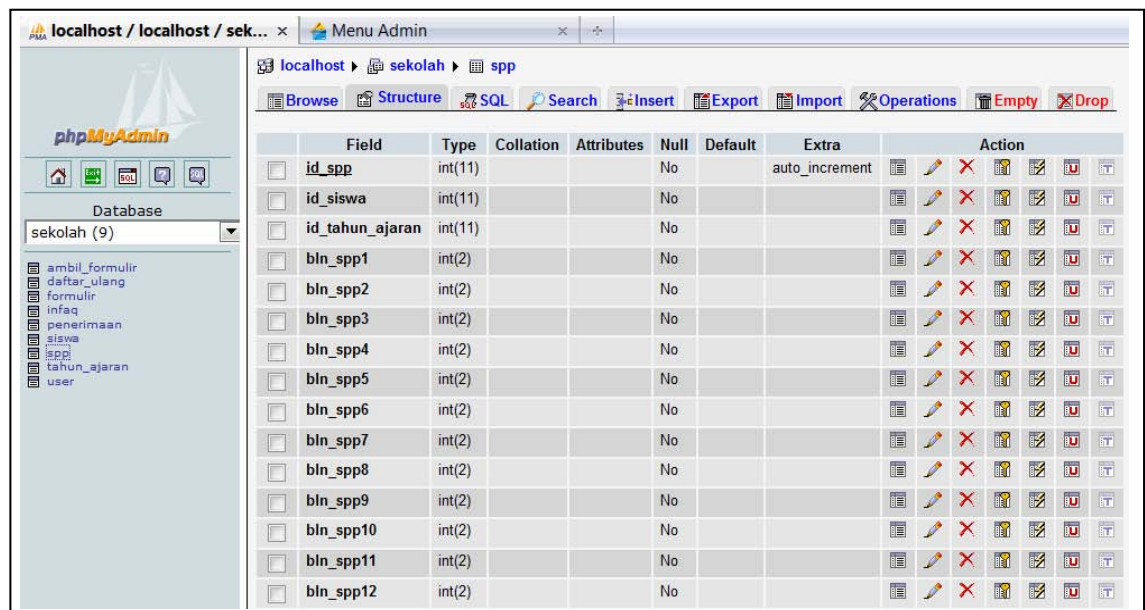
Gambar 4.15 merupakan database siswa. Database ini menyimpan data siswa yang sudah diterima dan sudah ditentukan kelasnya. Id_siswa merupakan primary_key dari database siswa, sedangkan id_penerimaan masuk sebagai foreign_key dimana database siswa membutuhkan id_penerimaan untuk memulai proses berlangsungnya pembagian kelas siswa.



Gambar 4.15. Database Siswa

7) Database SPP / Syahriah

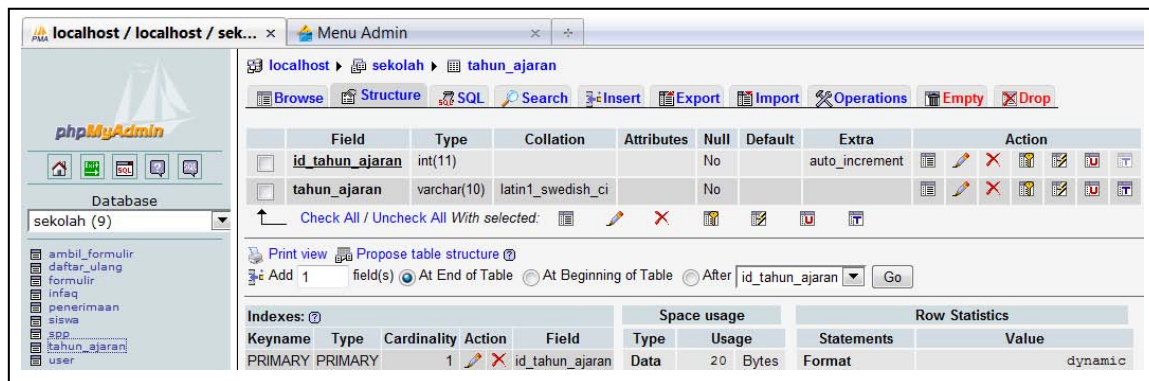
Gambar 4.16 merupakan database spp / syahriah. Database ini menyimpan data syahriah siswa yang membayar setiap bulannya. Id_spp merupakan primary_key dari database spp, sedangkan id_siswa dan id_tahun_ajaran masuk sebagai foreign_key dimana database spp membutuhkan dua foreign_key tersebut untuk memulai proses berlangsungnya pembayaran spp / syahriah siswa setiap bulan.



Gambar 4.16. Database SPP

8) Database Tahun Ajaran

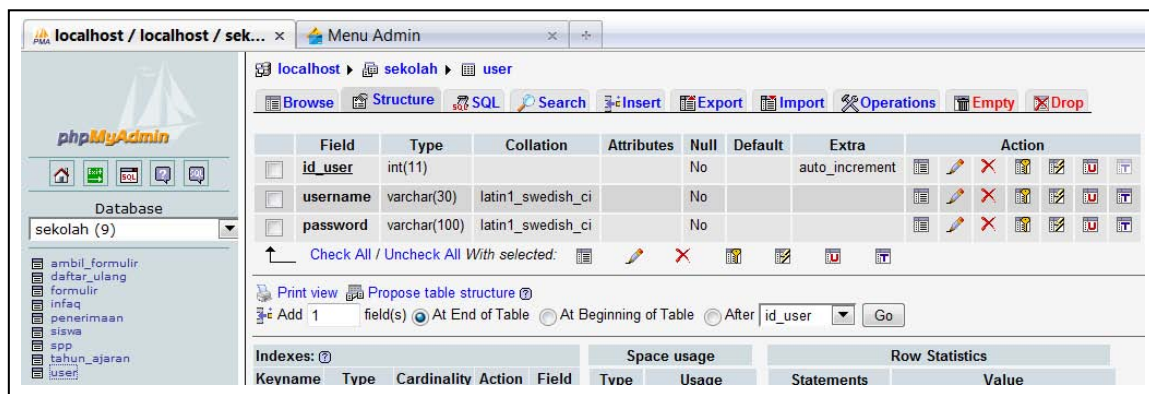
Gambar 4.17 merupakan database tahun ajaran. Database ini menyimpan data tahun ajaran baru setiap tahunnya. Id_tahun_ajaran merupakan primary_key dari database tahun ajaran, sedangkan database ini tidak memiliki foreign_key.



Gambar 4.17. Database Tahun Ajaran

9) Database User

Gambar 4.18 merupakan database user. Database ini menyimpan data user / admin. Id_user merupakan primary_key dari database user, sama halnya dengan database tahun ajaran, database ini tidak memiliki foreign_key.



Gambar 4.18. Database User

4.3 Implementasi Antar Muka

Pada tahap ini akan dijabarkan tentang implementasi antar muka dari aplikasi yang telah dibuat berdasarkan perancangan yang telah dibahas pada Bab III. System ini berfungsi untuk mempermudah admin dalam mengelola system administrasi pada SD Khadijah 2 Surabaya. Pada system ini terdapat beberapa sub menu utama yaitu :

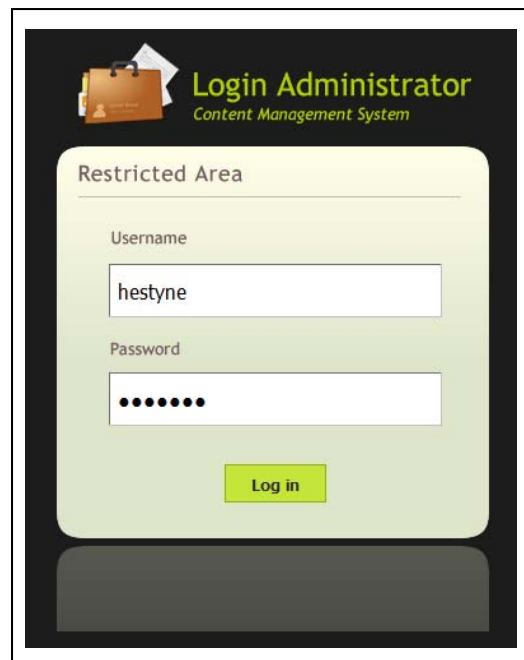
- 1) Form Login Admin
- 2) Halaman Utama
- 3) Menu Pendaftaran Siswa Baru :
 - a. Form Pengambilan Formulir
 - b. Form Formulir Pendaftaran
 - c. Form Penerimaan Siswa
- 4) Menu Pembayaran Siswa :
 - a. Form Pembayaran Syahriah
 - b. Form Pembayaran Infaq
- 5) Menu Registrasi Siswa :
 - a. Form Kelas 1
 - b. Form Kelas 2
 - c. Form Kelas 3
 - d. Form Kelas 4
 - e. Form Kelas 5
 - f. Form Kelas 6
- 6) Menu Rekapitulasi
 - a. Form Rekap Formulir

- b. Form Rekap Infaq
 - c. Form Rekap Syahriah
 - d. Form Rekap Daftar Ulang
- 7) Menu Amin
- 8) Logout

Pada halaman berikutnya akan dijelaskan satu per satu *form* yang ada di dalam program ini.

4.3.1 Form Login Admin

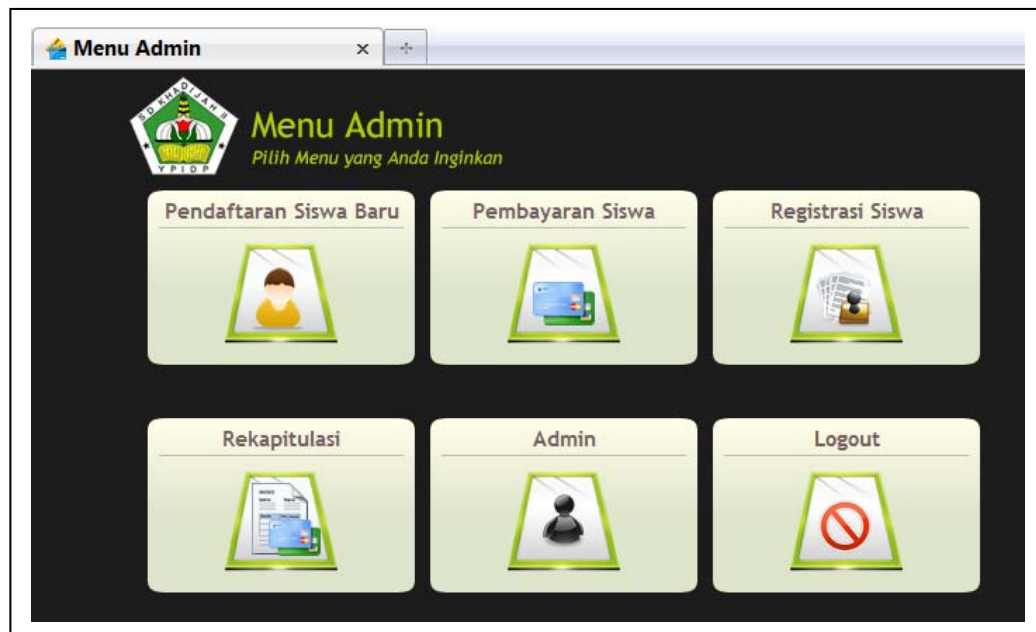
Gambar 4.19 ini merupakan sebuah proses awal sebelum masuk ke dalam menu utama dimana administrator harus melakukan proses login terlebih dahulu dengan memasukkan username dan passwordnya.

The image shows a web-based login interface. At the top, there is a logo of a folder with a document and the text 'Login Administrator' and 'Content Management System'. Below this is a light green box with a rounded top and a shadow. Inside this box, the text 'Restricted Area' is at the top. Below it are two input fields: 'Username' with the text 'hestyne' and 'Password' with a series of dots. At the bottom of the box is a yellow button with the text 'Log in'.

Gambar 4.19 Form Login Admin

4.3.2 Halaman utama

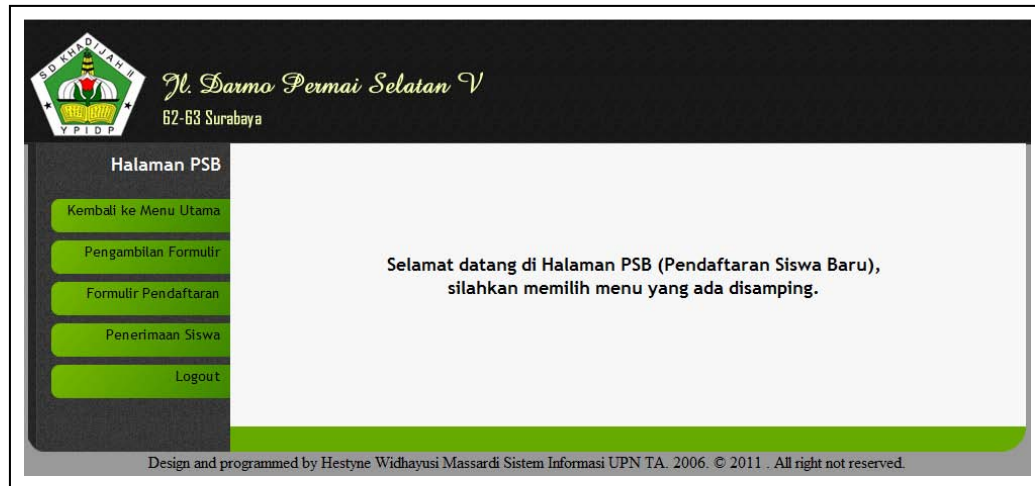
Gambar 4.20 ini merupakan tampilan menu utama yang keluar setelah admin melakukan proses login. Pada halaman menu utama ini terdapat beberapa sub-sub menu diantaranya adalah menu pendaftaran siswa baru, menu pembayaran siswa, menu registrasi siswa, menu rekapitulasi, menu admin dan logout.



Gambar 4.20. Halaman Utama

4.3.3. Menu Pendaftaran Siswa Baru

Gambar 4.21 merupakan halaman dari menu pendaftaran siswa baru. Pada halaman ini terdapat beberapa sub menu pilihan diantaranya kembali ke menu utama, pengambilan formulir untuk data pengambil formulir, formulir pendaftaran untuk mengisi formulir, penerimaan siswa untuk data siswa yang diterima dan logout untuk keluar dari login.



Gambar 4.21. Halaman Pendaftaran Siswa Baru

1) Form Pengambilan Formulir

Gambar 4.22 merupakan form pengambilan formulir. Dimana form ini berfungsi untuk menampilkan data pembeli / pengambil formulir. Untuk menambah data pembeli / pengambil formulir, admin dapat mengeklik link tambah data pada pojok kanan atas seperti gambar 4.22 di bawah ini.



Gambar 4.22. Form Pengambilan Formulir

2) Form Pendaftaran Siswa Baru

Gambar 4.23 merupakan form pendaftaran siswa baru. dimana form ini menyimpan data siswa yang menyerahkan formulir yang telah dibeli / diambil oleh orangtua siswa sebelumnya. Disini, admin memasukkan datanya harus sesuai dengan no.formulir yang telah diambil / dibeli sebelumnya. Gambar 4.23 terlihat jelas no.form yang ditunjukkan tidak berurutan.

Formulir Pendaftaran Siswa Baru

Formulir Pendaftaran Siswa Baru SD Khadijah II Surabaya Tahun Pelajaran 2010/2011

No. Form : [Cari](#) [Tambah Data](#)

No.Form	Nama	Status	Alamat	No.Telp	Action
2	Roedy Kartono	Orang Tua	Perum Graha Mitra Asri B6	0317563457	Ubah
3	Suwarno	Orang Tua	Dukuh Pakis 24	0315660010	Ubah
5	Sumardi, SE	Orang Tua	Donowati VII / 8	0317328765	Ubah
6	Kho, Djuliharto	Orang Tua	Dukuh Pakis I A / 48	081231421777	Ubah

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 4.23. Form Formulir Pendaftaran

3) Form Penerimaan Siswa

Gambar 4.24 merupakan form penerimaan siswa. Form ini menampilkan data siswa yang sudah diterima dan untuk memproses registrasi dan pembagian kelas. Apabila calon siswa telah diterima sebagai siswa baru, admin akan memproses dengan mengeklik action rincian. Dan siswa tersebut telah resmi menjadi siswa baru SD Khadijah 2 Surabaya.

No. Form	Nama Orang Tua	Nama Calon Siswa	Alamat	No. Telp	Infaq	Status	Action
2	Roedy Kartono	Firman Aero Naradika	Perum Graha Mitra Asri B6	0317563457	0	Belum Diproses	Rincian
3	Suwarno	Layyina Januar Ma'rifah	Dukuh Pakis 24	0315660010	0	Belum Diproses	Rincian
5	Sumardi, SE	Fairuz Rakha Permana	Donowati VII / 8	0317328765	0	Belum Diproses	Rincian
6	Kho, Djuliharto	Florencia	Dukuh Pakis I A / 48	081231421777	0	Belum Diproses	Rincian

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 4.24. Form Penerimaan Siswa

4.3.4. Menu Pembayaran Siswa

Gambar 4.26 merupakan halaman pembayaran dari menu pembayaran siswa. Pada halaman ini terdapat beberapa sub menu pilihan diantaranya kembali ke menu utama, pembayaran syahriah untuk pembayaran spp siswa setiap bulan, pembayaran infaq untuk pembayaran infaq siswa baru dan logout untuk keluar dari login.

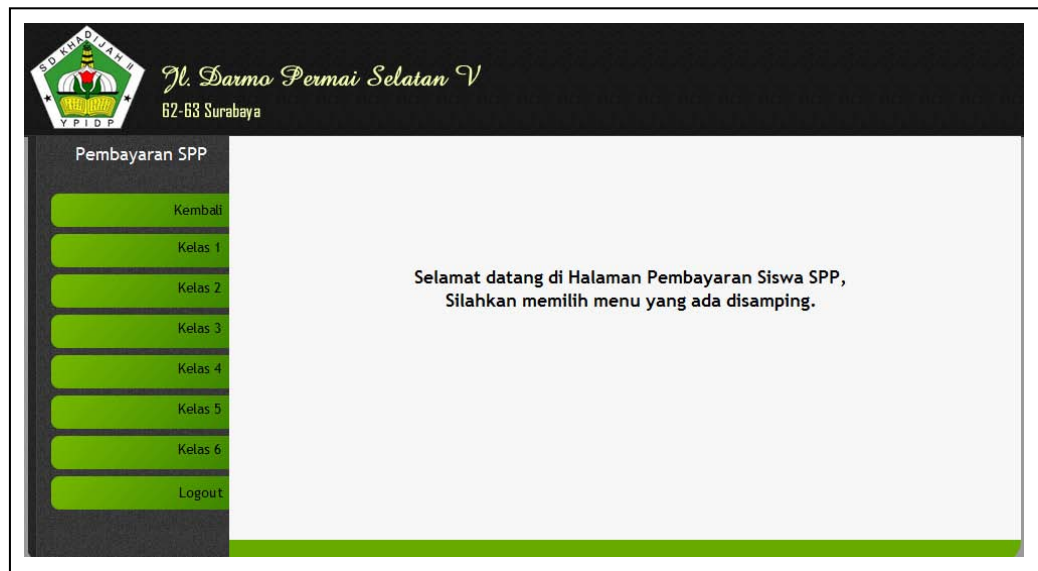
Selamat datang di Halaman Pembayaran Siswa,
Silahkan memilih menu yang ada disamping.

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 4.25. Halaman Pembayaran Siswa

1) Form Pembayaran Syahriah

Gambar 4.26 merupakan halaman pembayaran syahriah. Dimana dalam sub menu pembayaran syahriah terdapat pilihan kelas mulai dari kelas 1(satu) sampai dengan kelas 6(enam).



Gambar 4.26. Halaman Pembayaran Syahriah Siswa

Gambar 4.27 merupakan form pembayaran spp dari menu halaman pembayaran syahriah yang terdaftar pada kelas 1(satu) A. dalam form ini, terdapat nama siswa dan total syahriah yang belum terbayarkan. Dan apabila siswa telah membayar syahriah melalui bank, dan pihak bank juga telah mendebitkan , maka admin akan menyentang pembayaran syahriah dan toal syahriah yang belum terbayarkan akan berkurang.

Pembayaran Spp

Pembayaran SPP Kelas 1-A Tahun Pelajaran 2010/2011

Daftar Siswa

	Nama	Sisa Syariah per Tahun	SPP											
			01	02	03	04	05	06	07	08	09	10	11	12
1.	Layyina Januar Ma'rifah	3600000												
2.	Firman Aero Naradika	3600000												

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 4.27. Form Pembayaran Spp / Syahriah Siswa

2) Form Pembayaran Infaq

Gambar 4.28 merupakan form pembayaran infaq. Form ini menampilkan proses pembayaran infaq. Apabila siswa membayar infaq, maka admin akan memasukkan jumlah besaran infaq pada action rincian.

Pembayaran Infaq

Data Pembayaran Infaq Siswa SD KHADIJAH 2 Tahun Pelajaran 2010/2011

No	NIS	Nama Siswa	Kelas	Jumlah Infaq	Jumlah Dibayar	Action
1	1000	Firman Aero Naradika	1-A	0	0	Lunas
2	1001	Layyina Januar Ma'rifah	1-A	2.250.000	0	Rincian

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 4.28. Form Pembayaran Infaq

4.3.5. Menu Registrasi Siswa

Gambar 4.29 merupakan halaman daftar ulang dari menu registrasi siswa. Pada halaman ini terdapat pilihan menu mulai dari kelas 1(satu) sampai dengan kelas 5(lima) sedangkan kelas 6(enam) tidak perlu melakukan proses ini karena kelas 6(enam) dinyatakan telah keluar (lulus) dari SD Khadijah 2 Surabaya.

The screenshot shows a web application interface for student re-enrollment. The header is dark with a logo on the left and the school name 'N. Darma Permai Selatan V' and address '62-63 Surabaya' on the right. The sidebar on the left has a 'Daftar Ulang' button, a 'Kembali Ke Menu Utama' button, and a list of class selection buttons: 'Kelas 1', 'Kelas 2', 'Kelas 3', 'Kelas 4', 'Kelas 5', and 'Logout'. The main content area is light gray and contains the text: 'Selamat datang di Halaman Daftar Ulang Siswa, Silahkan memilih asal kelas siswa yang ada disamping.' The footer is a green bar with the text: 'Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.'

Gambar 4.29. Halaman Daftar Ulang

Gambar 4.30 merupakan form daftar ulang dari menu halaman daftar ulang. Form ini menentukan naik atau tidaknya siswa tersebut ke kelas berikutnya. Proses ini tidak dapat di proses apabila siswa tersebut belum melunasi pembayaran spp / syahriah selama 2(dua) semester atau 1(satu) tahun. Di dalam form daftar ulang ini terdapat kunci data, fungsi kunci data disini adalah untuk memindahkan para siswa yang naik kelas ke kelas berikutnya. Apabila semua siswa telah diproses kenaikan

kelasnya oleh admin. Apabila ada siswa yang belum membayar, maka admin dilarang untuk mengunci data.

Daftar Ulang

Kembali Ke Menu Utama

Kelas 1 1 - A

Kelas 2 1 - B

Kelas 3

Kelas 4

Kelas 5

Logout

Daftar Ulang Siswa Kelas 1-A Tahun Pelajaran 2010/2011

Daftar Siswa

No.	NIS	Nama	Naik Kelas		Ke Kelas	Action
			Ya	Tidak		
1.	1000	Firman Aero Naradika	<input checked="" type="radio"/>	<input type="radio"/>	2-A ▼	Simpan
2.	1001	Layyina Januar Ma'rifah	<input type="radio"/>	<input type="radio"/>	2-A ▼	Simpan

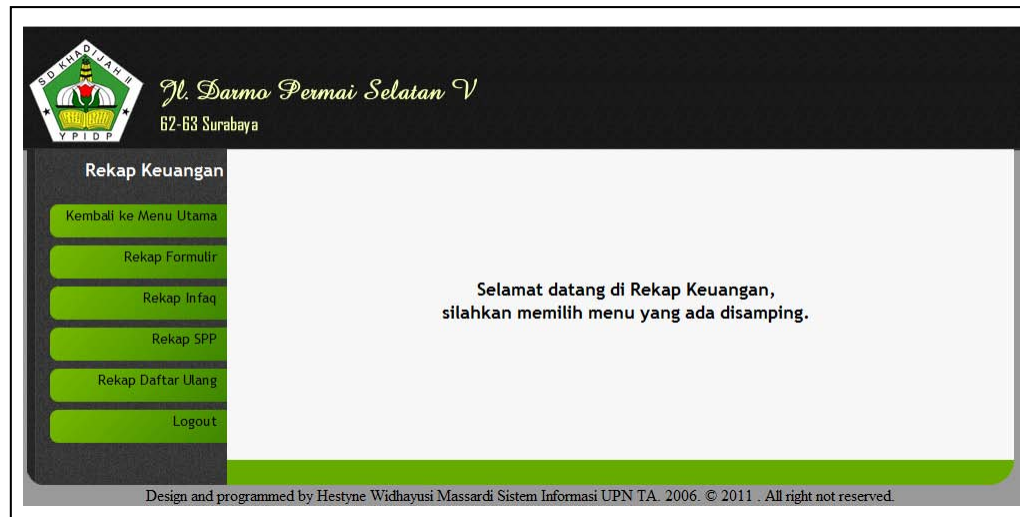
Kunci Data

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011. All right not reserved.

Gambar 4.30. Form Daftar Ulang

4.3.6. Menu Rekapitulasi


Gambar 4.31 merupakan halaman rekap keuangan dari menu rekapitulasi. Pada halaman ini terdapat pilihan menu diantaranya adalah kembali ke menu utama, form rekap formulir untuk melihat jumlah total rupiah yang telah mengambil / membeli formulir, form rekap infaq untuk melihat jumlah total rupiah siswa yang telah membayar infaq, form rekap spp untuk melihat siswa yang telah membayar spp dan siswa yang menunggak spp, form rekap daftar ulang untuk melihat jumlah rupiah siswa yang telah melakukan proses registrasi / daftar ulang. dan logout untuk keluar.



Gambar 4.31. Halaman Rekap Keuangan

1) Form Rekap Formulir

Gambar 4.32 merupakan form rekap formulir. Pada form ini menampilkan data pengambil formuli, jumlah rupiah, total rupiah yang didapat dari penjualan formulir, laporan dan grafik.

 U. Darmo Permai Selatan V 62-63 Surabaya							
Rekap Keuangan	Pengambilan Formulir						
Kembali ke Menu Utama	Laporan Pengambilan Formulir SD KHADIJAH 2 Tahun Pelajaran 2010/2011						
Rekap Formulir	<div>Laporan</div> <div>Grafik</div>						
Rekap Infaq	No.Form	Tgl.Ambil	Nama	Alamat	No.Telp	Dari TK	Jumlah
Rekap SPP	1	16-05-2011	Dedy Pramana	Simomulyo Baru 1G / 25	0317389342	Sabilillah	200.000
Rekap Daftar Uang	2	16-05-2011	Roedy Kartono	Perum Graha Mitra Asri B6	0317563457	Dharma Bakti	200.000
Logout	3	17-05-2011	Suwarno	Dukuh Pakis 24	0315660010	Dharma Wanita	200.000
	4	17-05-2011	Aris Hariyadi W	Manukan Lor II A / 140	0317408413	Sabilillah	200.000
	5	17-05-2011	Sumardi, SE	Donowati VII / 8	0317328765	Sabilillah	200.000
	6	18-05-2011	Kho, Djuliharto	Dukuh Pakis I A / 48	081231421777	Dharma Wanita	200.000

Gambar 4.32. Form Rekap Pengambilan Formulir

2) Form Rekap Infaq

Gambar 4.33 merupakan form rekap pembayaran infaq. Pada form ini menampilkan kelas 1(satu) sampai dengan kelas 6(enam) dimana terdapat jumlah infaq yang belum terbayarkan dan jumlah infaq yang sudah terbayarkan pada masing-masing kelas.

Kelas	Jumlah Infaq	Jumlah Dibayar
1	2.250.000	0
Jumlah	Rp.2.250.000	Rp.0

Gambar 4.33. Form Rekap Pembayaran Infaq

3) Form Rekap SPP

Gambar 4.34 merupakan form rekap pembayaran spp. Form ini menampilkan nama siswa, kelas siswa, syahriah siswa yang belum terbayarkan dan yang sudah terbayarkan, serta sisa syahriah. Pada bagian bawah form ini terdapat jumlah total rupiah syahriah dan pembagian syahriah senilai Rp.300.000 diantaranya Rp.10.000 untuk komite, Rp.35.000 untuk kegiatan dan sisanya untuk yayasan.

Rekap Keuangan Laporan Pembayaran SPP Tahun Pelajaran 2010/2011

Daftar Siswa

No.	Nama	Kelas	Syariah	Dibayar	Sisa
1.	Layyina Januar Ma'rifah	1-A	3.600.000	0	3.600.000
2.	Firman Aero Naradika	1-A	3.600.000	0	3.600.000
Jumlah			Rp.7.200.000	Rp.0	Rp.7.200.000

Komite : Rp.0
Kegiatan : Rp.0
Yayasan : Rp.0

Design and programmed by Hestvne Widhavusi Massardi Sistem Informasi UPN TA. 2006. © 2011. All right not reserved.

Gambar 4.34. Form Rekap Pembayaran SPP / Syariah

4) Form Rekap Daftar Ulang

Gambar 4.35 merupakan form rekap daftar ulang. Pada form ini menampilkan nama siswa yang telah melakukan proses daftar ulang (kenaikan kelas) serta biaya daftar ulang yang nantinya akan dijumlah total yang didapat dari biaya daftar ulang per siswa.

Rekap Keuangan Laporan Daftar Ulang Siswa Tahun Pelajaran 2010/2011

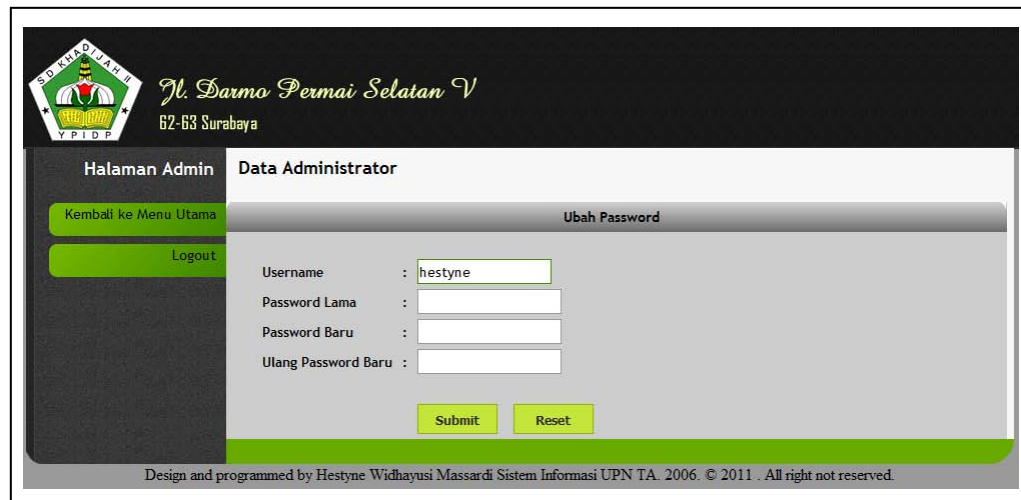
Daftar Siswa

No.	Nama	Dari Kelas	Ke Kelas	Ket	Total
1.	Cinta Kasih Putri Andini	1-A		-	-
2.	Ryan Wahyu Satrio	1-A	2-A	Naik	250.000
3.	Layyina Januar Ma'rifah	1-A		-	-
4.	Naura Alfi Amelia	1-B		-	-
5.	Nisa'ul Aulia	1-B		-	-
6.	Fairuz Rakha Permana	1-B		-	-
7.	Anggellina Sri Putri Sukma Purba Nur Imani	2-A		-	-
8.	Jessica Rubiansyah	2-A		-	-
9.	Heaven Ade Aldrico	2-A		-	-
10.	Eriana Sevita Sari	2-B		-	-

Gambar 4.35. Form Rekap Daftar Ulang

4.3.7. Menu Admin

Gambar 4.36 merupakan halaman admin dari menu admin. Dimana halaman ini berisi form penggantian password admin dari password lama diganti dengan password yang baru.

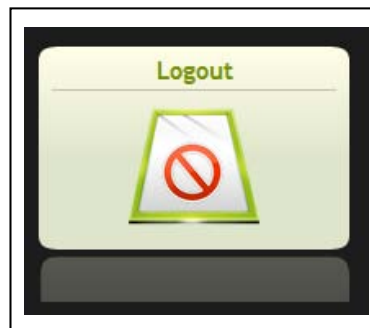


The screenshot shows a web application interface for an administrator. At the top left is a logo for 'SD KEMAJUAN YPIDP' and the text 'N. Darmo Permai Selatan V 62-63 Surabaya'. Below this is a sidebar with 'Halaman Admin' and two buttons: 'Kembali ke Menu Utama' and 'Logout'. The main content area is titled 'Data Administrator' and contains a form titled 'Ubah Password'. The form has four input fields: 'Username' (pre-filled with 'hestyne'), 'Password Lama', 'Password Baru', and 'Ulang Password Baru'. There are 'Submit' and 'Reset' buttons at the bottom of the form. A footer note reads: 'Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.'

Gambar 4.36. Form Admin

4.3.8. Logout

Gambar 4.37 merupakan tampilan logout. Dimana menu ini digunakan untuk keluar ketika admin sudah tidak memerlukan proses supaya aplikasi tidak disalahgunakan oleh pihak yang tidak bertanggungjawab.



Gambar 4.37. Tampilan Logout

BAB V

UJI COBA DAN EVALUASI

Pada bab ini akan dibahas mengenai uji coba terhadap aplikasi yang telah dibuat dan selanjutnya akan dibuat evaluasi dari hasil uji coba tersebut. Uji coba dilaksanakan untuk mengetahui apakah aplikasi dapat berjalan dengan baik sesuai perancangan yang dibuat. Evaluasi dilakukan untuk menentukan tingkat keberhasilan dari aplikasi tersebut.

5.1 Lingkungan Uji Coba

Pada uji coba kali ini akan diterapkan mulai dari proses pendaftaran siswa baru hingga pembayaran syahriahnya. Spesifikasi uji coba kali ini adalah sebagai berikut :

- a. Sistem Operasi : Windows 7
- b. Server Database PHPMyAdmin
- c. Web Server : Apache Triad
- d. Processor Intel Core 2 Duo
- e. Memori RAM 2 Gb
- f. Hardisk 160 GB

5.2 Skenario Uji Coba

Untuk memastikan bahwa aplikasi ini berjalan lancar, penyusun akan menyusun scenario, yang akan diuji coba, antara lain:

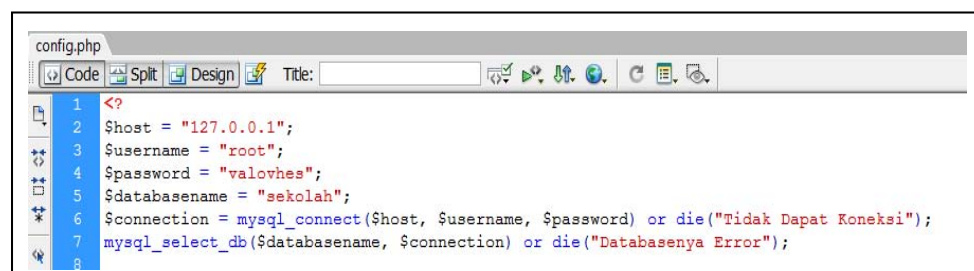
- a. Uji coba koneksi konfigurasi.
- b. Uji coba *login*, sebagai akses halaman pertama saat menjalankan aplikasi.
- c. Uji coba koneksi *database*, berfungsi untuk mengetahui apakah *database* bisa koneksi atau terjadi *error* pada waktu koneksi.
- d. Uji coba proses pendaftaran siswa baru
- e. Uji coba proses pembayaran siswa
- f. Uji coba proses registrasi siswa (daftar ulang)

5.3 Pelaksanaan Uji Coba

Pada sub-bab ini akan dijelaskan *step-by-step* mengenai pelaksanaan skenario uji coba yang telah dijabarkan pada sub-bab sebelumnya, untuk membuktikan uji coba aplikasi disertakan gambar tentang kejadian yang sedang berlangsung pada aplikasi system pendaftaran siswa baru dan pembayaran syahriah ini.

5.3.1 Uji Coba Koneksi Konfigurasi

Sebelum menjalankan aplikasi, terlebih dahulu melakukan koneksi konfigurasi antara program dengan database seperti Gambar 5.1 berikut.

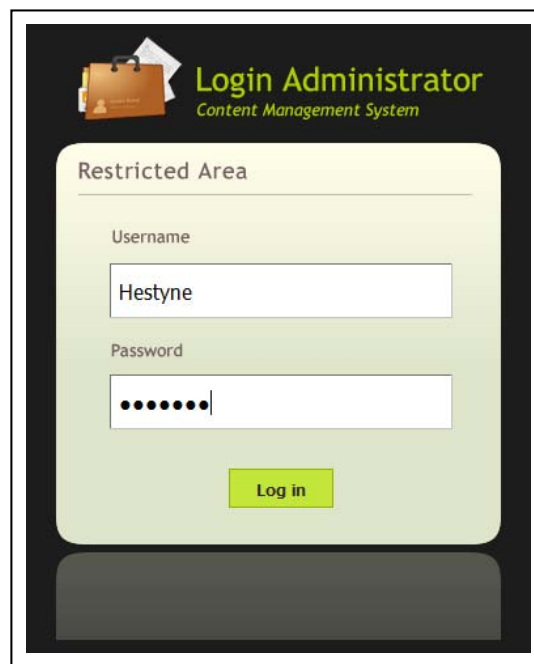


Gambar 5.1. Uji Coba Koneksi Konfigurasi

5.3.2 Uji Coba Login

Untuk mengakses aplikasi maka diperlukan login terlebih dahulu. Jika tidak melakukan login maka aplikasi tidak dapat diakses atau digunakan.

- 1) Gambar 5.2 adalah tampilan login. Pertama *admin* melakukan login terlebih dahulu dengan mengisi *username* dan *password* lalu tekan *login*.



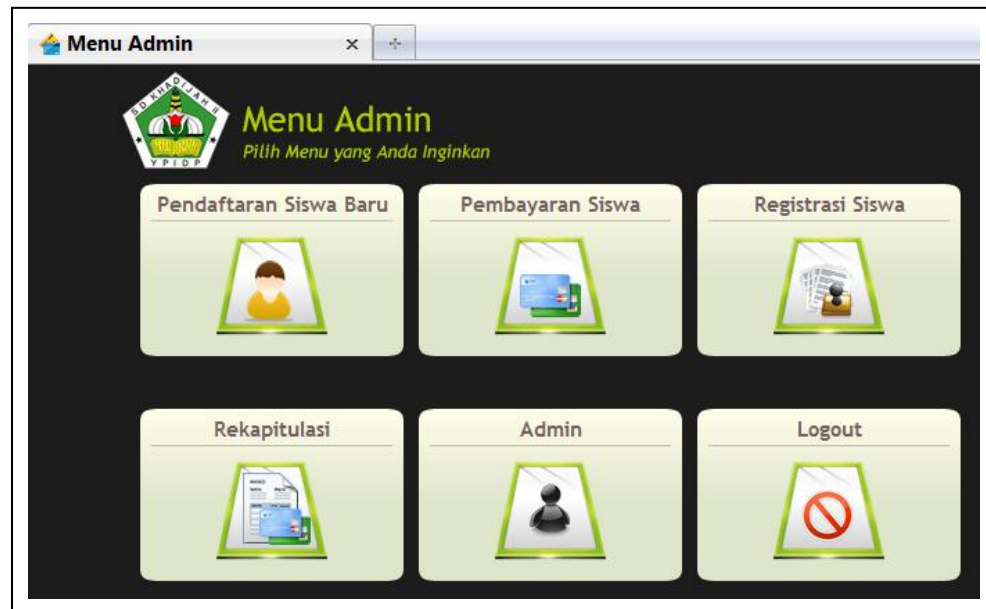
Gambar 5.2. Tampilan Login

- 2) Apabila terjadi kesalahan dalam pengisian username dan password, akan muncul warning seperti Gambar 5.3 dan admin harus mengulangi pengisian dengan benar.



Gambar 5.3. Warning Login

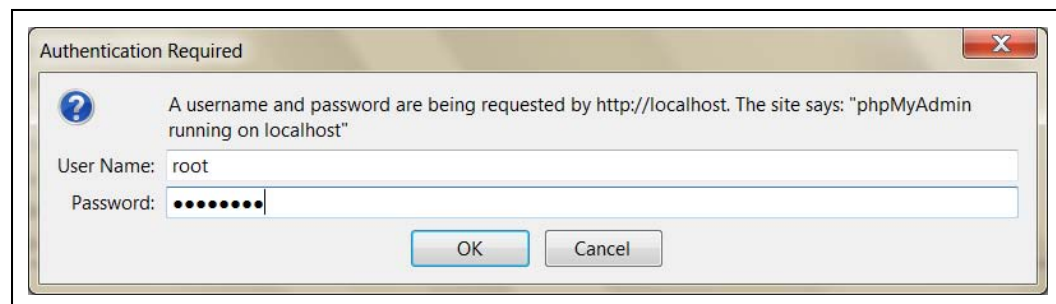
- 3) Setelah admin berhasil login, maka system akan masuk ke menu utama seperti Gambar 5.4 berikut.



Gambar 5.4. Halaman Menu Utama

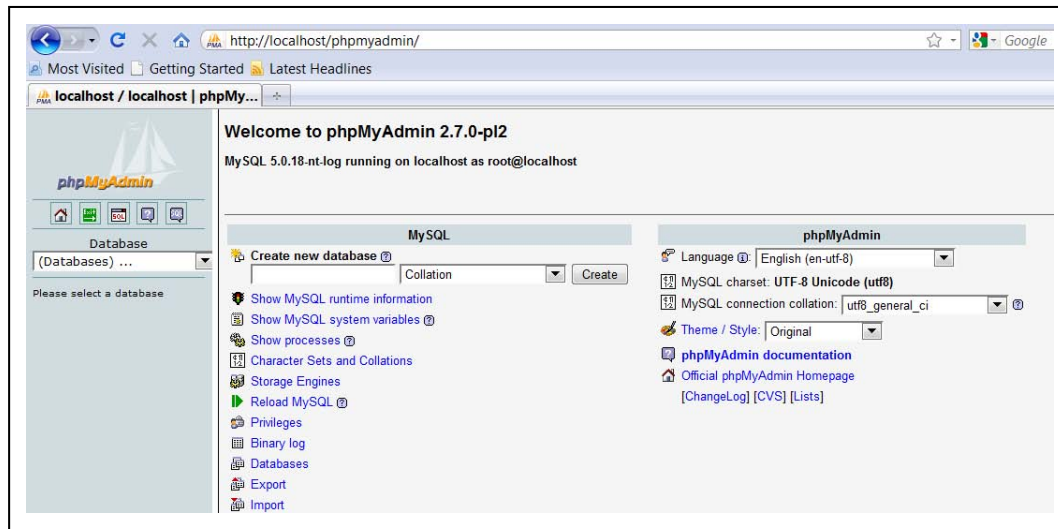
5.3.3 Uji Coba Koneksi *Database*

Uji coba ini untuk mengetahui apakah koneksi web server ke database berhasil atau terdapat error waktu koneksi seperti Gambar 5.5 berikut.



Gambar 5.5. Uji Coba Koneksi *Database*

Gambar 5.6 merupakan tampilan database PHPMyAdmin yang apabila konfigurasi dari program ke database berhasil dikoneksikan.



Gambar 5.6. Tampilan Database PHPMyAdmin

5.3.4 Uji Coba Proses Pendaftaran Siswa baru

Uji coba ini dilakukan untuk mengetahui apakah system yang dibuat untuk pendaftaran siswa baru bisa berjalan dengan baik atau masih ada *error*. Proses tersebut akan dijelaskan seperti pada step di bawah ini, dimulai dengan proses pengambilan formulir, formulir pendaftaran hingga penerimaan siswa.

1) Proses Pengambilan Formulir

Proses pengambilan formulir ini berfungsi untuk mendata pengambil yang membeli formulir dan untuk menjumlah berapa formulir yang habis terjual. Proses yang dilakukan admin ketika pengambil formulir membeli formulir adalah sebagai berikut :

- a. Setelah admin melakukan login dan muncul halaman utama, selanjutnya masuk ke dalam menu pendaftaran siswa, pilih menu pengambilan formulir

dan klik tambah data kemudian isi data pengambil formulir dengan benar seperti Gambar 5.7 berikut, lalu klik tombol submit.

Halaman PSB **Pengambilan Formulir**

Kembali ke Menu Utama

Pengambilan Formulir

Formulir Pendaftaran

Penerimaan Siswa

Logout

Form Pengambilan Formulir SD KHADIJAH 2 Tahun Pelajaran 2010/2011

No. Form :

Nama :

Alamat :

No. Telepon :

Dari TK :

Gambar 5.7. Form Isi Data Pengambil Formulir

- b. Untuk mengubah data pengambil, lakukan proses pengubahan data pengambil formulir dengan mengklik action ubah seperti pada Gambar 5.8 berikut. Maka akan tampil kembali seperti pada Gambar 5.7 dan lakukan perubahan data dengan benar.

Halaman PSB **Pengambilan Formulir**

Kembali ke Menu Utama

Pengambilan Formulir

Formulir Pendaftaran

Penerimaan Siswa

Logout

Form Pengambilan Formulir SD KHADIJAH 2 Tahun Pelajaran 2010/2011

No. Form : [Tambah Data](#)

No. Form	Tgl. Ambil	Nama Pengambil	Alamat	No. Telp	Dari TK	Action
1	16-05-2011	Dedy Pramana	Simomulyo Baru 1G / 25	0317389342	Sabillillah	Ubah
2	16-05-2011	Roedy Kartono	Perum Graha Mitra Asri B6	0317563457	Dharma Bakti	Ubah
3	17-05-2011	Suwarno	Dukuh Pakis 24	0315660010	Dharma Wanita	Ubah
4	17-05-2011	Aris Hariyadi W	Manukan Lor II A / 140	0317408413	Sabillillah	Ubah
5	17-05-2011	Sumardi, SE	Donowati VII / 8	0317328765	Sabillillah	Ubah
6	18-05-2011	Kho, Djuliharto	Dukuh Pakis I A / 48	081231421777	Dharma Wanita	Ubah
7	19-05-2011	Dadang Iskandar	Dukuh Kupang XVI / 50	0315683732	Dharma Bakti	Ubah

Gambar 5.8. Ubah Data Pengambil Formulir

2) Proses Formulir Pendaftaran

Proses formulir pendaftaran ini berfungsi untuk memasukkan data siswa yang telah mengisi formulir dengan lengkap dan mengembalikannya pada petugas / admin. Pihak admin memasukkan data pada form formulir pendaftaran ini sesuai dengan no.form pendaftaran. Proses yang dilakukan admin dalam memasukkan data siswa yang telah mengembalikan formulir adalah sebagai berikut :

- a. Setelah admin melakukan login dan muncul halaman utama, selanjutnya masuk ke dalam menu pendaftaran siswa, pilih menu formulir pendaftaran dan klik tambah data kemudian isi data siswa dengan benar sesuai dengan no.form yang ada di pengambilan formulir seperti Gambar 5.9 berikut, lalu klik tombol submit.

Gambar 5.9. Form Pendaftaran Siswa

- b. Untuk mengubah data pendaftaran siswa yang sudah ada, lakukan proses pengubahan data formulir pendaftaran dengan mengklik action ubah seperti

pada Gambar 5.10 berikut. Maka akan tampil kembali seperti pada Gambar 5.9 dan lakukan perubahan data dengan benar

Halaman PSB **Formulir Pendaftaran Siswa Baru**

Kembali ke Menu Utama
Pengambilan Formulir
Formulir Pendaftaran
Penerimaan Siswa
Logout

Formulir Pendaftaran Siswa Baru SD Khadijah II Surabaya Tahun Pelajaran 2010/2011

No. Form : [Tambah Data](#)

No.Form	Nama	Status	Alamat	No.Telp	Action
2	Roedy Kartono	Orang Tua	Perum Graha Mitra Asri B6	0317563457	Ubah
3	Suwarno	Orang Tua	Dukuh Pakis 24	0315660010	Ubah
5	Sumardi, SE	Orang Tua	Donowati VII / 8	0317328765	Ubah
6	Kho, Djuliharto	Orang Tua	Dukuh Pakis I A / 48	081231421777	Ubah

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 5.10. Ubah Data Pendaftaran Siswa Baru

3) Proses Penerimaan Siswa

Proses penerimaan siswa ini berfungsi untuk memproses siswa yang telah diterima dengan memasukkan gelombang 1(satu) atau gelombang 2(dua) sesuai dengan batas waktu gelombang yang telah ditentukan pihak sekolah serta penentuan ruang kelas dan no.induk siswa. Proses yang dilakukan admin dalam memproses data siswa yang telah diterima adalah sebagai berikut :

- a. Setelah admin melakukan login dan muncul halaman utama, selanjutnya masuk ke dalam menu pendaftaran siswa, pilih menu penerimaan siswa. Maka akan muncul tampilan form seperti Gambar 5.11 berikut.

No. Form	Nama Orang Tua	Nama Calon Siswa	Alamat	No. Telp	Infaq	Status	Action
2	Roedy Kartono	Firman Aero Naradika	Perum Graha Mitra Asri B6	0317563457	0	OK	Rincian
3	Suwarno	Layyina Januar Ma'rifah	Dukuh Pakis 24	0315660010	2.250.000	OK	Rincian
5	Sumardi, SE	Fairuz Rakha Permana	Donowati VII / 8	0317328765	0	OK	Rincian
6	Kho, Djuliharto	Florencia	Dukuh Pakis I A / 48	081231421777	4.500.000	OK	Rincian

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 5.11. Form Penerimaan Siswa

- b. Apabila admin ingin merubah status siswa, klik action rincian untuk mengelompokkan gelombang dan kelas serta memasukkan no.induk siswa seperti Gambar 5.12 berikut, lalu klik tombol submit.

No. Form : 2

Nama Orang Tua : Roedy Kartono

Alamat : Perum Graha Mitra Asri B6

No. Telepon : 0317563457

Nama Calon Siswa : Firman Aero Naradika

Gelombang : ☒ I ☐ II

Diterima Di Kelas : 1-A

Nomor Induk Siswa : 1000

Gambar 5.12. Form Rincian Penerimaan Siswa

- c. Gambar 5.13 merupakan hasil dari proses perubahan status dari belum diproses menjadi ok. Pada form ini terlihat jelas infaq tiap siswa berbeda-beda dikarenakan jika siswa tersebut anak kandung pertama dari guru dapat diskon infaq 100%. Jika siswa tersebut anak kandung kedua dari guru juga mendapat diskon 50%.

No. Form	Nama Orang Tua	Nama Calon Siswa	Alamat	No. Telp	Infaq	Status	Action
2	Roedy Kartono	Firman Aero Naradika	Perum Graha Mitra Asri B6	0317563457	0	OK	Rincian
3	Suwarno	Layyina Januar Ma'rifah	Dukuh Pakis 24	0315660010	2.250.000	OK	Rincian
5	Sumardi, SE	Fairuz Rakha Permana	Donowati VII / 8	0317328765	0	OK	Rincian
6	Kho, Djuliharto	Florenia	Dukuh Pakis I A / 48	081231421777	4.500.000	OK	Rincian

Gambar 5.13. Hasil dari Proses Penerimaan Siswa

Source code di bawah ini merupakan rumus fungsi dari pemisahan diskon pembayaran infaq untuk anak guru dan bukan anak guru.

```
function infaq($val1, $val2, $val3, $val4){
    if($val1==1 or $val2==1){
        if($val3>0 and $val3<=2){
            if($val4==1) $jml = 4000000;
            else if($val4==2) $jml = 4500000;
            if($val3==1) $infaq = $jml - ((100/100)*$jml);
            else if($val3==2) $infaq = $jml - ((50/100)*$jml);
        }else {
            if($val4==1) $infaq = 4000000;
            else if($val4==2) $infaq = 4500000;
        }
        return $infaq;
    }
}
```

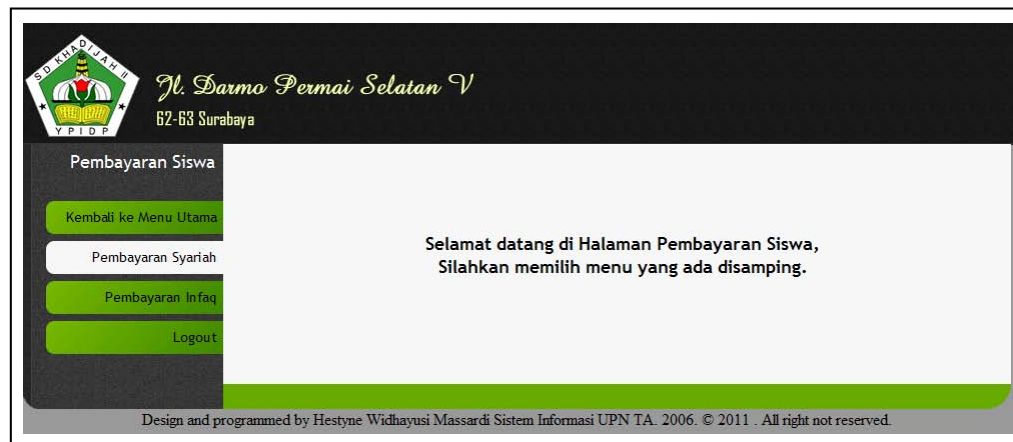
5.3.5 Uji Coba Proses Pembayaran Siswa

Uji coba ini dilakukan untuk mengetahui apakah system yang dibuat untuk pembayaran siswa bisa berjalan dengan baik atau masih ada *error*. Proses tersebut akan dijelaskan seperti pada proses di bawah ini, dimulai dengan proses pembayaran syahriah serta proses pembayaran infaq.

1) Proses Pembayaran Syahriah

Proses pembayaran syahriah ini berfungsi untuk mempermudah admin untuk memproses pembayaran syahriah siswa yang diterima dari bank. Proses yang dilakukan admin ketika melakukan proses pembayaran syahriah adalah sebagai berikut :

- a. Setelah admin melakukan login dan muncul halaman utama, selanjutnya masuk ke dalam menu pembayaran siswa, pilih menu pembayaran syahriah seperti Gambar 5.14 berikut..



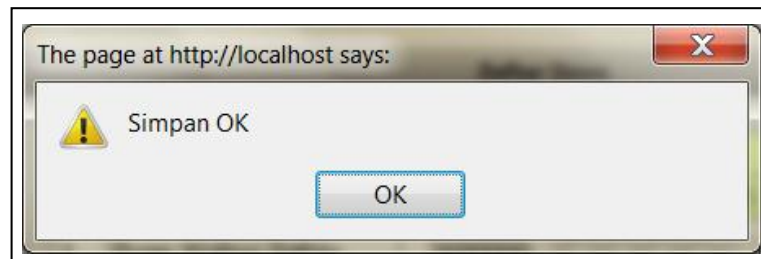
Gambar 5.14. Halaman Pembayaran Syahriah

- b. Setelah memilih menu pembayaran syahriah, maka system akan masuk pada pemilihan kelas seperti Gambar 5.15 berikut, kemudian centang pembayaran spp. Maka akan muncul alert laporan simpan OK seperti pada Gambar 5.16.

	Nama	Sisa Syariah per Tahun	SPP											
			01	02	03	04	05	06	07	08	09	10	11	12
1.	Layyina Januar Ma'rifah	3600000	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2.	Firman Aero Naradika	3600000	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 5.15. Form Pembayaran SPP



Gambar 5.16. Laporan Simpan Berhasil

- c. Setelah berhasil maka biaya spp akan berkurang secara otomatis seperti pada Gambar 5.17 berikut.

Pembayaran Spp

Pembayaran SPP Kelas 1-A Tahun Pelajaran 2010/2011

Daftar Siswa

No.	Nama	Sisa Syariah per Tahun	SPP											
			01	02	03	04	05	06	07	08	09	10	11	12
1.	Layyina Januar Ma'rifah	3000000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2.	Firman Aero Naradika	3300000	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 5.17. Hasil dari Proses Pembayaran Spp

2) Proses Pembayaran Infaq

Proses pembayaran infaq ini berfungsi untuk memproses pembayaran infaq siswa. Proses yang dilakukan admin ketika melakukan proses pembayaran infaq adalah sebagai berikut :

- Setelah admin melakukan login dan muncul halaman utama, selanjutnya masuk ke dalam menu pembayaran siswa, pilih menu pembayaran infaq seperti Gambar 5.18 berikut..

Pembayaran Siswa

Pembayaran Infaq

Data Pembayaran Infaq Siswa SD KHADIJAH 2 Tahun Pelajaran 2010/2011

No	NIS	Nama Siswa	Kelas	Jumlah Infaq	Jumlah Dibayar	Action
1	1000	Firman Aero Naradika	1-A	0	0	Lunas
2	1001	Layyina Januar Ma'rifah	1-A	2.250.000	250.000	Rincian
3	1002	Fairuz Rakha Permana	1-B	0	0	Lunas
4	1003	Florencia	1-B	4.500.000	0	Rincian

Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.

Gambar 5.18. Halaman Form Pembayaran Infaq

- b. Apabila ada siswa yang membayar infaq, maka klik action rincian pada nama siswa yang membayar infaq, maka akan keluar form seperti pada Gambar 5.19. kemudian masukkan nominal yang dibayarkan siswa dan submit.

Gambar 5.19. Form Pengisian Infaq

- c. Setelah berhasil mengisi pembayaran infaq, akan muncul perubahan biaya infaq seperti Gambar 5.20 berikut. Apabila jumlah infaq 0 (nol) maka secara otomatis system akan menjadi lunas.

No	NIS	Nama Siswa	Kelas	Jumlah Infaq	Jumlah Dibayar	Action
1	1000	Firman Aero Naradika	1-A	0	0	Lunas
2	1001	Layyina Januar Ma'rifah	1-A	2.250.000	250.000	Bancian
3	1002	Fairuz Rakha Permana	1-B	0	0	Lunas
4	1003	Florencia	1-B	4.500.000	250.000	Bancian

Gambar 5.20. Form Perubahan Data Infaq

5.3.6 Uji Coba Proses Registrasi Siswa

Uji coba ini dilakukan untuk mengetahui apakah system yang dibuat untuk registrasi siswa bisa berjalan dengan baik atau masih ada *error*. Proses tersebut akan dijelaskan seperti pada Gambar 5.21 di bawah ini :

- a. Pilih nama siswa naik kelas yang melakukan proses registrasi dengan membayar biaya registrasi sebesar Rp. 250.000,-. Pilih kelasnya dan simpan. Dalam proses ini, tidak diperbolehkan mengunci data apabila masih ada siswa yang belum melakukan registrasi / daftar ulang.

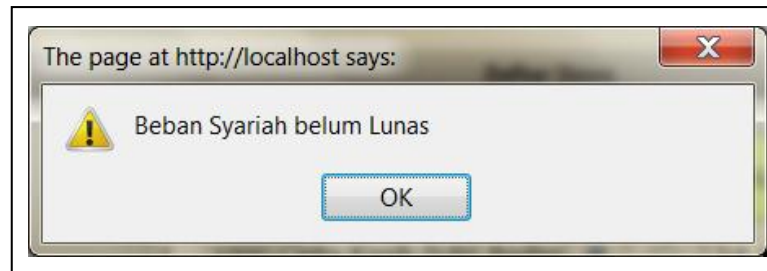
The screenshot shows a web application interface for 'Daftar Ulang' (Re-enrollment). The header includes the logo of 'SEKOLAH YPIDP' and the text 'Jl. Darmo Permai Selatan V 62-63 Surabaya'. The main content area is titled 'Daftar Ulang Siswa Kelas 1-A Tahun Pelajaran 2010/2011'. On the left, there is a sidebar with a 'Daftar Ulang' button and a list of classes (Kelas 1, 2, 3, 4, 5) and a 'Logout' button. The main area contains a table with the following data:

No.	NIS	Nama	Naik Kelas		Ke Kelas	Action
			Ya	Tidak		
1.	1000	Firman Aero Naradika	<input checked="" type="radio"/>	<input type="radio"/>	2A	Simpan
2.	1001	Layyina Januar Ma'rifah	<input type="radio"/>	<input type="radio"/>	2B	Simpan

Below the table is a 'Kunci Data' button. At the bottom, there is a footer text: 'Design and programmed by Hestyne Widhayusi Massardi Sistem Informasi UPN TA. 2006. © 2011 . All right not reserved.'

Gambar 5.21. Halaman Form Daftar Ulang

- b. Proses ini tidak dapat dilakukan apabila siswa belum melunasi seluruh pembayaran spp. Seperti pada Gambar 5.22 yang muncul ketika siswa tersebut belum melunasi seluruh pembayaran syahriah



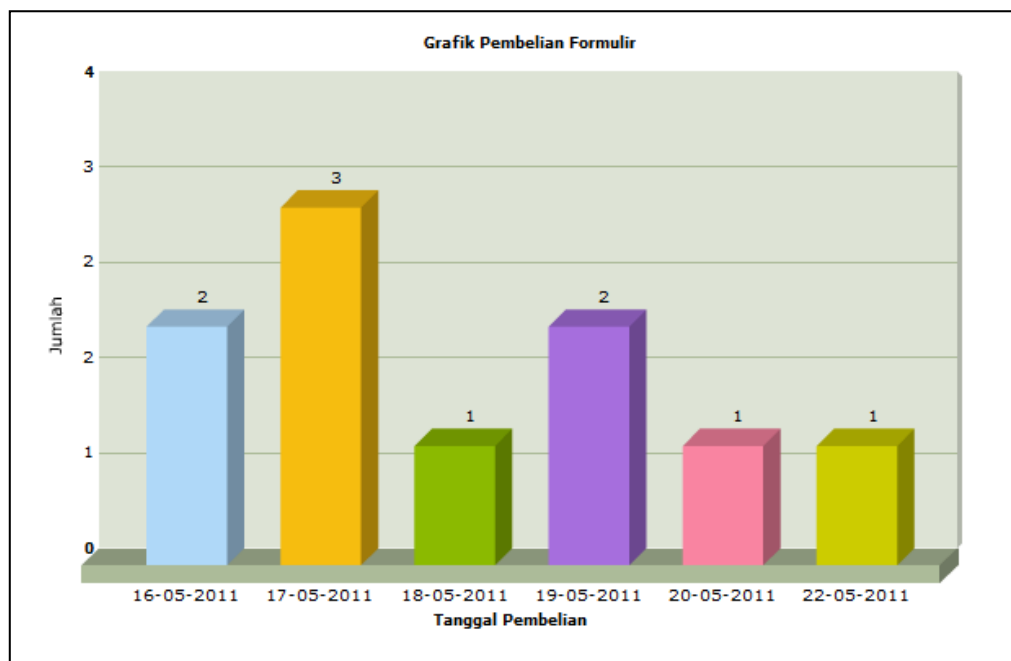
Gambar 5.22. Warning Syahriah Belum Lunas

5.4 Grafik Rekapitulasi

Berikut merupakan gambar-gambar grafik yang diperoleh dari transaksi pembayaran sebelumnya.

a. Grafik Rekap Formulir

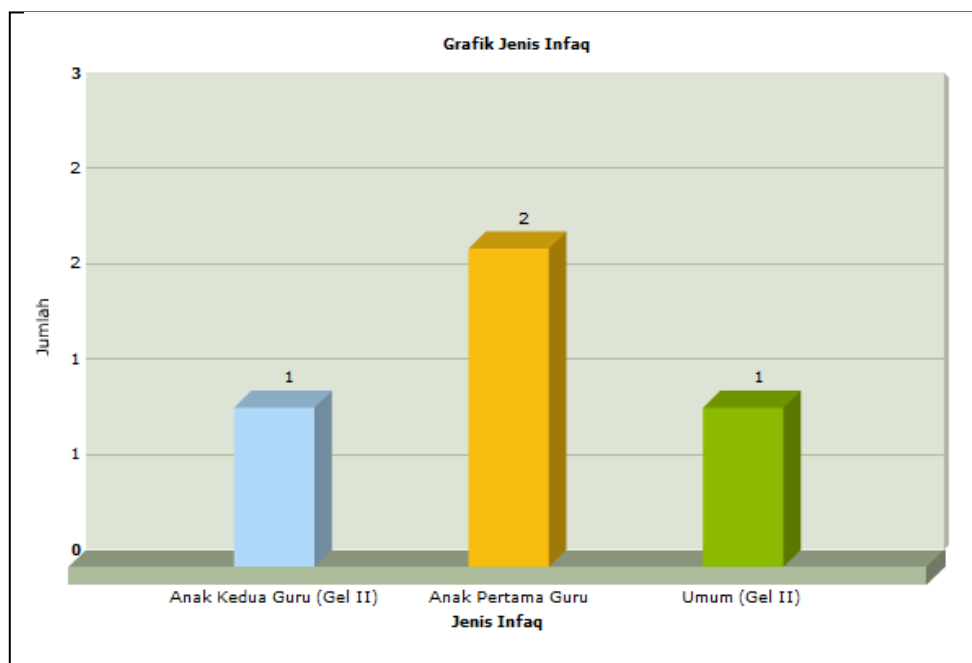
Gambar 5.23 merupakan grafik rekap pembelian formulir. Dimana grafik ini menunjukkan transaksi tanggal pembelian dan jumlah yang membeli formulir pendaftaran.



Gambar 5.23. Grafik Rekap Formulir

b. Grafik Rekap Infaq

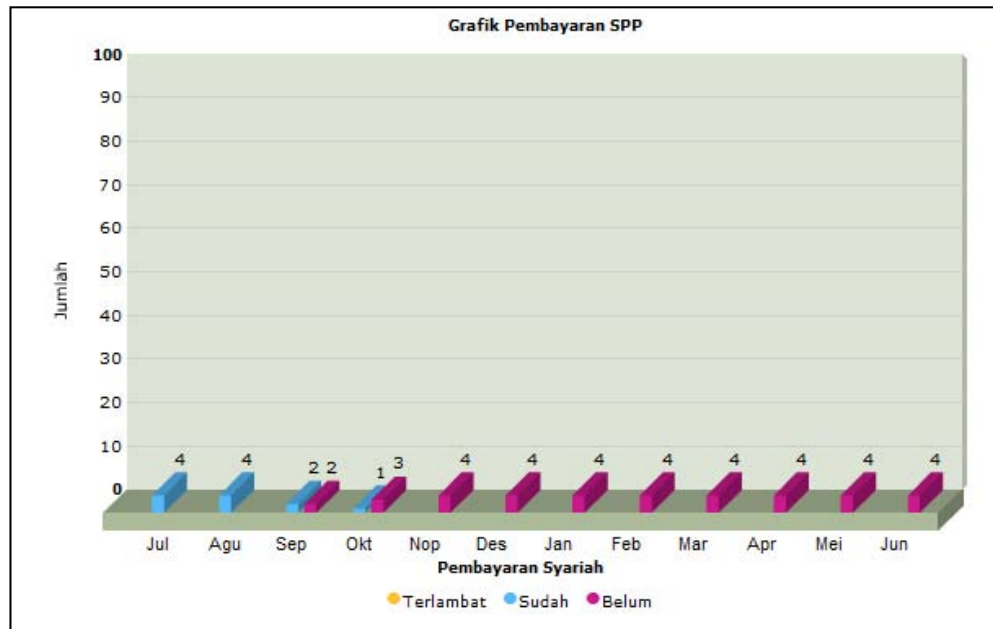
Gambar 5.24 merupakan grafik rekapitulasi infaq. Dimana grafik ini menunjukkan pembagian infaq bagi siswa yang anak pertama dan kedua Guru untuk mendapatkan potongan pembayaran infaq sesuai dengan gelombang pendaftarannya.



Gambar 5.24. Grafik Rekap Infaq

c. Grafik Rekap Spp/Syahriyah

Gambar 5.25 merupakan grafik rekap syahriyah/spp siswa. Grafik ini menampilkan grafik pembayaran siswa yang sudah membayar dan yang belum membayar serta yang terlambat membayar.



Gambar 5.25. Grafik Rekap Spp/Syahriyah

5.4 Evaluasi

Hasil evaluasi dari uji coba terhadap beberapa scenario yang telah dilakukan membuktikan bahwa aplikasi yang telah berjalan dengan baik sesuai dengan fungsinya masing-masing. Hanya saja proses ini dilakukan secara berurutan. Proses ini tidak dapat berjalan dengan baik apabila, penginputan data tidak sesuai dan tidak berurutan.

BAB VI

PENUTUP

6.1 Kesimpulan

Dari uraian diatas, diperoleh suatu kesimpulan mengenai program dan laporan tugas akhir ini, antara lain sebagai berikut :

- a. System informasi administrasi ini dapat menjadikan data yang semula diproses secara manual, menjadi terkomputerisasi.
- b. Untuk membuat rancangan system informasi ini dengan cara menyusun tabel-tabel database yang diperlukan oleh pihak sekolah dan merancang konsep program serta desain sesuai dengan kebutuhan yang dibutuhkan pihak sekolah. System ini telah di aplikasikan serta memenuhi harapan SD Khadijah 2 Surabaya.

6.2 Saran

Berdasar pengalaman dalam prosesa pembuatan aplikasi ini, terdapat beberapa saran yang diusulkan oleh penyusun terkait untuk pengembangan aplikasi lebih lanjut. Saran-saran tersebut antara lain:

- a. Dengan banyaknya data siswa yang ada di SD Khadijah 2 Surabaya, system ini masih memanggil keseluruhan data yang ada di database.
- b. Untuk pergantian tahun ajaran baru dan penambahan data admin masih dilakukan secara manual melalui database.
- c. Apabila semua data siswa telah terkunci pada menu daftar ulang, maka data tidak dapat di operasikan kembali jika ada siswa yang baru melakukan proses daftar ulang.

DAFTAR PUSTAKA

- Al Fatta, Hanif, 2007 , *"Analisis dan Perancangan Sistem Informas"*, Penerbit Andi, Jakarta.
- H, Rafiza, 2004 , *"Panduan dan Referensi Kamus Fungsi Php 5 untuk Membangun Database Berbasis Web"*, Elex Media Komputindo, Yogyakarta.
- Lukmanul Hakim, 2004, *"Membongkar Trik Rahasia Para Master PHP Lukmanul Hakim"*, Yogyakarta.
- Nugroho Bima, 2004, *"PHP & MySQL Dengan Editor Dreamweaver MX"*, penerbit ANDI, Yogyakarta.
- Prasetyo, Didik Dwi 2007 , *"Belajar Sendiri Tanpa Guru Php dan MySQL"*, Elex Media Komputindo, Yogyakarta.
- Sunarfrihantono, 2003, *"PHP & MySQL untuk Web"* Penerbit ANDI, Yogyakarta.
- Sutarman, 2006, *"Pemrograman Web Php dan MySQL"*, Graha Ilmu, Yogyakarta.